Johns Hopkins University

GUIDANCE TO THE FUTURE DIRECTION: CREATING A CLOUD-BASED REPOSITORY USING AWS

by
MINSOO KIM

A capstone project submitted to the
Krieger School of Arts and Sciences
Advanced Academic Programs
Johns Hopkins University
in partial fulfillment of the Degree of
Master of Science in Research Administration

Baltimore, Maryland
December 2019

Abstract

The profound scope of research administration brings an administrative burden, and the administrative burden takes a toll on faculties conducting the research. Almost half of the time spent on research is distributed to the research-related administrative tasks. Drowning in the increase of administrative burden, institutions have implemented a solution to reduce administrative burden and increase the productivity in research administration: a cloud-based infrastructure. These cloud-based infrastructures are created using commercial products and services provided by various vendors in the market. While there are institutions that utilize a cloud-based infrastructure to automate, simplify, and reduce administrative tasks, many institutions have yet to understand the need for it and adopt a cloud-based infrastructure.

This project provides a tutorial guide for institutions of higher education, without a cloud-based infrastructure, to create a cloud-based repository. The cloud-based repository in the guide is created from a cloud platform provided by Amazon Web Services (AWS). Unlimited storage, cost optimization, and automated process for repetitive tasks, cloud-based repositories take the administrative burden off of faculties.

Conclusively, the guide can be used by institutions to implement a customized cloud-based repository suited to their needs. The guide addresses the post-award phase of the award lifecycle. However, the cloud-based repository can be improved and enhanced to handle the workloads in the complete lifecycle of awards: pre-award, award, and post-award.

Table of Content

Figures

Abbreviations

AWS	Amazon Web Services
CDN	Content Delivery/Distribution Network
F&A Costs	Indirect Costs, also referred to as Facilities and Administrative Costs
Glacier	Referring to S3 Glacier
IHE	Institutions of Higher Education
S3	Simple Storage Service
TCO	Total Cost of Ownership

Glossary

Cloud-based.	On-demand computer system resources, such as applications, services, and storage, for users to access via the Internet.
Cloud-based Repository.	Storage service accessed by users using a cloud-based server.
Cloud Infrastructure.	Hardware and software components that makeup cloud computing.
Content Distribution Network.	Geographically distributed network that delivers or distributes content to various locations.
Cost Optimization.	Ensuring the maximized output for the minimum input required to perform the necessary tasks.
Elasticity.	The ability for the computing system to continually increase and decrease in size to compete with the demanded workload.
Fault Tolerance.	The ability for the computing system to function without delay even if the portion of the system has been impacted.
High Availability.	The ability to create numerous backups of the computing system.
Indirect Costs.	Financial expenditures relevant to the sponsored projects that cannot be directly associated with individual projects. Examples of Indirect Costs include the cost of electricity, administrative services, and usage of facilities. Indirect Costs are additionally known as Overhead or Facilities and Administration (F&A) Costs.
Lifecycle Policy.	Feature that allows modification of the lifespan of stored objects.
Post-Award Phase.	The last phase in the lifecycle of an award, which includes implementation, reporting, and closeout.
Record Management Service.	The storage service that the University of Washington offers to store documents in the post-award phase.
Simple Storage Service (S3).	Object storage designed to store and access any data over

	the Internet.[1]
S3 Glacier.	Secure, durable, and extremely low-cost Amazon S3 cloud storage classes for data archiving and long-term backup.[2]

[1] Lee Perlitz and Steven G. Elliott. "The Products." Amazon. Pearson Education Australia, 2000. https://aws.amazon.com/products/storage/?nc2=h_ql_prod_st.
[2] Ibid

Chapter 1. Introduction

1.1. Background.

The current practice of research administration requires a substantial amount of responsibilities from the research administrators. Research administrators are invested in not only assisting institution's faculty members in obtaining funding for research, but they are also devoted to ensuring that institutions are compliant with regulations that come with the funding. The competition for receiving funding has increased over the years. Additionally, the needs demanded by the sponsors are shifting over time. Institutions are compliant with varying regulations by different sponsors to ensure their relationships are in good standing. However, research administrators are burdened by an abundance of tasks that need to be fulfilled to maintain such relationships. The effort to reduce administrative burdens for research administration has become vital in the efficiency of institutions. Therefore, this capstone project addresses a record management system that is challenging the practice of research administration post-award phase. Specifically, focusing on the post-award phase, the capstone project dives into the following problems: the compliance guideline in terminating records, harmonization of Federal regulations, and reducing administrative burden.

1.2. Statement of the Problem.

Upon closing out an award, researchers are required to retain the records from the research activity for a certain amount of period. The period required for retention varies depending on the sponsor's requirements. Once the period for recordkeeping has expired, documents from the sponsored projects must be discarded. The award cycle comes to an end when the documents from the research activity are discarded upon expiration.

Institutions have implemented various record management services suited to their needs. The record management services, under the scope of research administration, requires physical space to retain documents and discard them when expired.

The University of Washington utilizes physical storage and appoints the responsibilities to the research administrators to store and destroy documents per state and federal regulations as well as sponsor requirements. These records need to be destroyed when expired to comply with various regulations. Records retained during the required period of time are subject to audit. Records retained beyond the record retention time period, if available, are subject to recall when there is an audit or internal investigation. Records destroyed in a timely manner in accordance with federal, state and local regulations are not available for audits and investigations. Thus, the IHE is at a reduced risk of audit findings if the records are destroyed at the appropriate time. Such responsibility for record destruction can be automated, utilizing a software-based infrastructure to store the documents away and discard them when they are expired. Instead of increasing the workload of research administrators, utilizing a software-based cloud repository could quickly alleviate such tasks.

1.3. Project Questions.

The author of this project addressed five questions:

1. What is the estimated financial cost for using a cloud-based infrastructure for a record management system?
2. How difficult is it for an institution to create and utilize a cloud-based infrastructure?

3. What are some of the issues that result from the failure to destroy the records according to state and federal laws?
4. Can institutions manage cloud-based infrastructures without paying the vendors?
5. Could this cloud-based infrastructure be implemented universally across all institutions within the United States?
6. Would it be possible to share the research data with other institutions by using the cloud-based infrastructure for a record management system?
7. How financially taxing would it be for institutions to migrate over to the cloud-based infrastructure?

1.4. Project Objectives.

This project was designed to explore the benefits of using a cloud repository, specifically Amazon Web Services (AWS) based system, for the record management service in financial, administrative, and compliance perspectives. By doing so, to guide the future direction of research administration in order to reduce the administrative burdens, administrative costs, and risks of non-compliance with Federal regulations. Thus, this project has the following objectives:

1. An instruction manual on how to create a cloud-based repository using AWS. This instruction manual can be used as a tutorial, and institutions can further customize it to suit their needs.

2. An example strategy that institutions can utilize to migrate from a physical repository to a cloud-based repository.

3. Illustrate the long-term benefits – financial, administrative, and compliance-related – of using a cloud-based repository for IHEs that still utilize physical capacities for the record management system.

4. Depict the potential of using a single cloud platform to create a universal cloud-based infrastructure that institutions across the United States can implement.

1.5. Significance.

The project is significant for institutions, especially institutions of higher education (IHE), to increase the efficiency of research administrators and to reduce burdens to store and destroy documents. Furthermore, utilizing less physical storage capacity can reduce the Indirect (F&A) costs. Lowered F&A costs would allow researchers to utilize more of the funding received from the sponsors. Furthermore, when institutions expand facilities, institutions can focus on utilizing capacities for purposes other than physically storing documents. In addition, the human error of losing the stored documents can be reduced even further by abiding by the guidance created from the project. Ultimately, the cloud-based system for record management system could be an initial step towards creating a universal platform for institutions to share research data in encryption.

1.6. Exclusions and Limitations.

The project is intended to explore why using the cloud-based infrastructure repository is an excellent future direction for the research administration. Therefore, the project will guide how to create a cloud-based infrastructure. This guide will work as general guidance and is not intended for a specific institution. Rather than applying every potential state and federal laws that institutions must abide, the guide will depict how the laws can be adapted to the cloud-based repository. There are other software solutions that

institutions can utilize to create a cloud-based infrastructure other than the AWS. The estimated usage cost is an example; thus, the actual costs of using the cloud-based repository may differ per use case. It is challenging to create a universal comparison between the current institutional structure and cloud-based repository since institutions differ from one another.

Chapter 2. Literature Review

2.1. Overview of Literature Review.

This project provides a guide for IHEs to migrate from a physical record management system to a cloud-based repository. Thus, the literature review dives into the current usage of cloud-based infrastructure in research administration. The need for a transition into a cloud-based infrastructure in research administration stems from the massive administrative burden. To reduce the administrative burden, IHEs have been implementing a solution: a cloud-based infrastructure. The case studies of successful implementation of cloud-based infrastructure are additionally examined to understand the benefits of a cloud-based infrastructure. Understanding the significance of cloud-based infrastructure in research administration, stakeholders of research administration in IHEs understand the potential and need for a cloud-based infrastructure further. Lastly, the literature review examines the case example – provided by a software solutions vendor Cayuse – of a successful implementation of cloud-based infrastructure by the University of California, San Diego.

2.2. Details of Review.

First, the comprehension of the magnitude of administrative burden in research administration takes priority. To understand the magnitude of the administrative burden, the Federal Demonstration Partnership (FDP) conducted a survey. The FDP is comprised of federal agencies and institutions of higher education that receive federal sponsorship. The FDP's mission is to find a way to reduce administrative burdens in sponsored research, specifically federally-sponsored research. The FDP survey was designated to understand the magnitude of administrative burdens placed on faculty during the

research. In the survey, 23,325 full-time faculty members participated in answering "questions on the nature, size, and impact of the administrative tasks associated with their research projects."[3] The survey illustrated over a quarter of the participants expressing a very significant amount of administrative burden on the participants.[4] Additionally, the FDP survey showed that "42% of the time spent by an average PI on a federally funded research project was reported to be expended on administrative tasks related to that project rather than on research."[5] This means that the average PI spends almost half of the time performing administrative tasks instead of focusing on the research. PIs are being federally-sponsored to conduct the research. Therefore, to expend the effort more on the research rather than the administrative tasks related to the research, institutions are implementing new solutions to enhance the effort expenditure.

To address the administrative burden, which reduces efficiency and diverts effort from the research, institutions utilize commercial software products and services to support research administration.[6] Commercial software products and services used by institutions to support research include PeopleSoft by Oracle, Workday Cloud by Workday, and Cayuse SP by Cayuse. The research administration covers the scope of the award lifecycle, which starts from the identification of funding opportunities to closing out an award. The award lifecycle is divided into three segments: pre-award, award, and post-award phase. Institutions have different systems integrated to address different phases of the award lifecycle. Therefore, to migrate from a physical capacity to a cloud-

[3] Sara Rockwell. "The FDP Faculty Burden Survey." Research management review. U.S. National Library of Medicine, 2009. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2887040/.
[4] Ibid
[5] Ibid
[6] Tyler Saas, James Kemp, Deloitte Consulting LLP, It Takes an Eco-System: A review of the Research Administration Landscape, Research Management Review, Volume 22, Number 1 (2017)

based capacity, institutions have opted for various software solutions to handle different needs in phases of the award lifecycle.

The University of San Diego, according to the case study provided by Cayuse, struggled with the piling number of administrative tasks, such as proposals and approvals, because they were done in the form of paperwork.[7] Since the administrative tasks were done on paper, research administrators had to take trips to offices, fill out the forms by hand, and hand-deliver everything to various departments within the university. Also, the University of San Diego did not have an electronic system established to browse through websites for funding opportunities. The number of administrative tasks was piling up, and the administrative burden did not seem to decrease. Thus, the institution utilized commercial software solutions by Cayuse (Cayuse424 and Cayuse SP) that provided "an electronic proposal development and submission solution along with … a sponsored project lifecycle management solution."[8]

The implementation of the cloud-based software brought benefits that impacted the institution vastly. The simplified proposal development and submission process allowed researchers to submit proposals while traveling, track the progress of the submitted proposals, and review budgets in a centralized system. Since the cloud-based software is a centralized system, faculties were all on the same page about the progress of a sponsored project lifecycle. Before the implementation of cloud-based infrastructure, the University of San Diego had merely 67 submissions. However, during the first year after the implementation of cloud-based infrastructure, the institution had about 115

[7] "University of San Diego Case Study." Cayuse. Accessed October 25, 2019. https://cayuse.com/case-study/university-san-diego-case-study/.
[8] Ibid

submissions.[9] Based on case studies highlighted on the Cayuse website, it is evident that the impact of a cloud-based infrastructure impacts research administration significantly. Migrating paperwork to electronic documents has made the administrative tasks more comfortable to process. Additionally, faculties were able to share the relevant information regarding proposals and sponsored project lifecycle across the institution on-demand.

It is not just Cayuse that has had a successful implementation case studies of a cloud-based system at an institution. PeopleSoft, a software-based solution by Oracle, has also been successfully adopted by numerous institutions. The University of Houston and the University of California, San Francisco, have adopted cloud-based infrastructure to alleviate the administrative burden on their institutions via PeopleSoft as well.[10, 11] Specifically, PeopleSoft Enterprise Grants Management software, "an entirely web-based solution that manages the full life cycle of research administration,"[12] was used to assist with billing, contracts, general ledger, grants, project costing, and receivables in the institutions.

As illustrated in the literature review and case studies, institutions have been benefitting from implementing a cloud-based infrastructure at their institutions. Institutions have been able to reduce administrative tasks and simplify the sponsored project lifecycle management. The reduced administrative burden has allowed faculties to focus more on the research than the administrative tasks related to the research.

---

[9] Ibid
[10] "The Research Administration System." The Research Administration System | Controller's Office. Accessed October 25, 2019. https://controller.ucsf.edu/how-to-guides/contracts-grants-accounting/research-administration-system.
[11] "PeopleSoft Grants." PeopleSoft Grants - University of Houston, Last modified September 26, 2019. https://www.uh.edu/research/sponsored-projects/peoplesoft/.
[12] "PeopleSoft Enterprise Grants Management." Oracle. Accessed October 25, 2019. http://www.oracle.com/us/products/applications/peoplesoft-enterprise/service-automation/peoplesoft-grants-management-065800.html.

2.3. Applicability of Literature Review.

While it is evident through case studies and literature review that a cloud-based infrastructure is beneficial to institutions, many have yet to implement and adopt the integration of a cloud-based infrastructure. Furthermore, there are numerous options for commercial software-based solutions that institutions can opt to implement based on their needs. Institutions may not feel the need for change despite the increasing workload and distracting efforts from the research. However, such may not be the case if stakeholders understand the purpose of the cloud-based infrastructure and the benefit from the implementation of a cloud-based infrastructure. Since the project focuses on the post-award phase of the award lifecycle, the cloud-based repository can reduce the administrative burden in retaining and managing documents that arose from the research activity. These retained documents have to be discarded when expired in order for institutions to be compliant with sponsors' requirements. A cloud-based repository can automate the retaining and discarding process, and allow faculties to expend more effort elsewhere. Additionally, a cloud-based repository does not take up physical space in the facility so that the institution can utilize the additional space for a more purposeful objective. Therefore, this project provides a guide that allows institutions to create a cloud-based repository that can be customized to fulfill the required needs.

Chapter 3. Need Assessment

3.1. Need Assessment.

While there are institutions that have implemented a cloud-based repository, many IHEs have yet to adopt a cloud-based repository. Therefore, the project provides a handbook on how to migrate from a physical record management services to a cloud-based repository. Institutions are already utilizing a cloud-based infrastructure to address needs such as efficiency, cost-optimization, and automation. Cayuse and PeopleSoft are two examples of software-based solutions that institutions utilize currently. These solutions are being used by institutions to address needs in the pre-award, post-award, and proposal submission process. Researchers want to devote their time and effort to the research itself rather than the required and increased administrative tasks. Therefore, institutions are putting in the effort to reduce administrative burdens and trying to automate the process.

As previously mentioned, PeopleSoft and Cayuse are examples of implemented solutions that mitigate the administrative burden and reduce administrative costs. These software-based solutions offer cloud-based repository solutions that institutions can adapt. However, many institutions still utilize physical capacity to store documents and discard the documents in person. If the record management system were migrated to a software-based repository, there would be less workforce required for research administrators to store and discard the documents physically. Not to mention, utilizing a software-based repository would reduce the costs of the institutions, resulting in a reduced F&A cost for researchers.

3.1.1 Assessment of Need.

Currently, no literature addresses the need for migrating physical storage capacities to cloud-based repositories. However, there are case studies provided by software-based solutions such as Cayuse that explain the benefits of using a cloud-based infrastructure. The case studies were very limited in a number of institutions utilizing a cloud-based infrastructure. Plus, the case studies did not address how the institution could migrate from an existing physical capacity to a cloud-based repository. Also, most of the institutions focused on using a cloud-based infrastructure to address the proposal submission process and pre-award phase of the research administration. Thus, this project focused on the post-award phase of the research administration that could benefit from the cloud-based infrastructure that institutions are already using. Therefore, it is essential to understand the benefit of using a cloud-based repository. Furthermore, this project guides the reader on how to migrate from a physical record management service to a cloud-based repository.

3.1.2. Metrics.

The following metrics used to establish the need: the cost required to store and discard documents in the AWS S3 (Simple Storage Service); potential reduction in F&A costs for researchers by utilizing a cloud-based repository rather than a physical facility; and reduction in work hours by research administrators to store and discard documents.

Institutions need a motivation to migrate from a current physical facility to a cloud-based repository. The motivation includes cost reduction and automation that can help institutions spend financial resources more on the research instead of administration. Therefore, understanding how much institutions save on the money by utilizing a cloud-based repository is an essential metric. Also, institutions can visualize the cost of creating

the cloud-based repository. Then, the institutions can perform a cost comparison of creating and maintaining the cloud-based repository versus the indirect costs of maintaining a physical facility. Furthermore, when institutions decide to expand, institutions reduce financial expenditure on building a facility space to keep the documents from research activity.

Additionally, an automated process for a cloud-based repository to store the documents and discard them when they are due should be measured. The metric for the automated process can be measured for the reduced work hours by research administrators. Once the cloud-based repository has been established, institutions can then compare the time it requires PIs to upload the document via the Internet versus research administrators physically filing all the documents and discarding them when expired.

3.2. Sources.

To establish the need for the project an Associate Vice Provost for Research Administration and Integrity, at the University of Washington, was consulted. To further assess the need for the University of Washington, which does not have a cloud-based repository system in place, the Director of Record Management Services was consulted.

Chapter 4. Project Description

4.1. Discussion of Project Elements.

This project involves a tutorial guide to creating a cloud-based repository using the AWS for IHE that have yet to implement a cloud-based repository solution. The tutorial provides instruction on how to migrate from an existing physical record management system to a cloud-based repository. The guide is solely based on using services provided by AWS because AWS is the most widely used cloud platform in the market currently. Additionally, AWS has products and services that institutions can utilize to additionally implement other cloud-based solutions to address different phases of the award lifecycle. Instead of using different vendors' solutions to meet the needs of institutions, institutions can utilize a single cloud platform that provides solutions that can meet every demand.

The tutorial starts with the basics of how to create an extensive cloud-based repository that can store all types of documents from the research activity. Explicitly, the guide incorporates AWS S3 to create the repository. AWS S3 is a service known to provide the most cost-efficient and unlimited amounts of storage. The tutorial guide continues with instructions on how to upload documents on to the repository and establish a lifecycle policy of the uploaded documents. Once the lifecycle policy has been established, the guide illustrates an example of a successfully discarded document. To ensure that the uploaded documents are kept private from external parties, the tutorial gives the user an option to make the entire repository private. Despite the privately existing cloud-based repository, the tutorial shows how the research administrators can grant access to the cloud-based repository for other faculty members of the institution. By

limiting access to the cloud-based repository, institutions can reduce the cost spent on utilizing the cloud-based repository since AWS charges the users based on the usage of the services and products, not an establishment.

In the case where the faculty may need to share the uploaded documents for auditing or internal control purposes, the guide demonstrates the simplicity in the distribution of the repository contents. Lastly, the guide includes some of the tools provided by the AWS free-of-charge to keep track of expenses. The tools can be used to set a budget on the total expenditure spent on the cloud-based repository and establish a threshold for the budget of the total expense.

Chapter 5. Methodology

5.1. Methodology Overview.

In order to conduct and complete the project, the project carries out the entire tutorial guide of creating a cloud-based repository. The tutorial guide is created to have the flexibility to adapt different institutional needs to migrate from a physical record management system. To understand some of those needs, the Director of the Record Management Service and the Associate Vice Provost of the Office of Research at the University of Washington were consulted. The University of Washington (UW) still utilizes a physical record management system, even though it is one of the largest public institutions in the US west coast.

The University of Washington has three separate campuses spread throughout the Seattle vicinity. Therefore, the Director of Record Management Services at UW provided the required needs that should be fulfilled for an institution to migrate from a physical record management system to a cloud-based repository. Meanwhile, the Associate Vice Provost was able to elaborate administrative burdens that come along with the current system at the institution. The cloud-based repository was created to address the needs provided by the Director of Records Management Service and the Associate Vice Provost of the Office of Research.

Furthermore, the project refers to case studies of institutions that have migrated to cloud-based infrastructure to improve the effort of research administration. The methods used in case studies by other institutions utilize different software-based solutions. Institutions can utilize other services, other than S3, in AWS to create a cloud-based repository. However, this project uses AWS S3 as the primary tool to create a cloud-

based project because it is known to be a limitless and cost-efficient service that AWS has to offer. Ultimately, rather than seeking various solutions by multiple vendors, institutions could utilize a universal cloud platform and implement numerous solutions to fulfill the varying needs in the future.

Lastly, the documents that have been uploaded as samples reflect various sizes and types of documents from research activity. The documents are in forms of pdf, images, videos, text files, and audios. The uploaded documents in the cloud-based repository illustrate the full compatibility of a cloud-based repository that institutions can migrate from a physical record management system.

5.2. Project Design and Discussion.

The design of the project was based on the diverse needs of institutions. Specifically, institutions that have yet to implement a cloud-based repository for the record management system. The project utilizes AWS as the sole provider for cloud-based infrastructure. First, the project was designed to be a guidance for IHEs that have a physical record management system or have yet to implement a cloud-based system. Therefore, system integration of the cloud-based repository was prioritized. The project utilizes images of the step-by-step cloud-based repository creation process to be user-friendly.

Second, the project provides instructions on how to upload copies of physical documents to a cloud-based repository. The documents are uploaded into a folder called buckets. These buckets can be categorized per research activity, and the lifecycle policy can be assigned to these buckets. The lifecycle policy assigned to a bucket can be used repeatedly on other buckets upon creation. The tutorial guide provides an example image

of a discarded bucket from the cloud-based repository when expired to prove the effectiveness of the lifecycle policy.

Furthermore, to illustrate the complete control of user access, the guide shows how to manage the users that have access to the cloud-based repository. The cloud-based repository is created privately, so without permission, the users cannot access and use the cloud-based repository. Since the repository is internally controlled, the external parties do not have access to the documents uploaded in the cloud-based repository as well. However, these external parties can receive permission to view the uploaded documents for various purposes: research collaboration, internal auditing, and compliance checks. Lastly, the guide allows the research administrator to access the financial aspect of using the cloud-based repository. The research administrator can view the cost usage and set the budget on the cloud-based repository usage. By doing so, the user can view the financial benefit of using the cloud-based repository rather than the physical facility.

Chapter 6. Project Results and Discussion

6.1. Project Results 1. Tutorial Guide to Create a Cloud-based Repository Using AWS S3

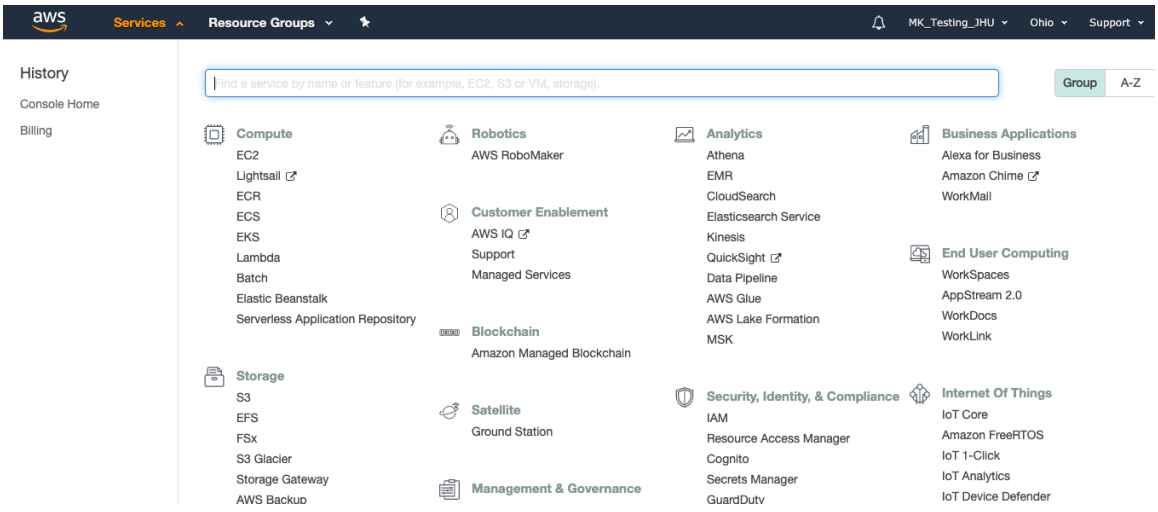
The outcome of this project is a tutorial guide for institutions to create a cloud-based repository using the services and products by the AWS. The guide is not targeted for a specific institution, but it is designed to be adapted by many institutions that do not have a cloud-based infrastructure. The guide can be used as a trial for institutions before fully migrating over to a cloud-based infrastructure. Overall, the tutorial illustrates how to set up the services to create a cloud-based repository, manage access to users that will utilize the repository, distribute the uploaded documents in the repository, and financially manage the repository.

The Amazon Web Services (AWS) is a cloud platform, most widely used, that provides customizable solutions and products. Any individuals and entities can easily use the services. Therefore, the guide utilizes the AWS cloud platform to create a cloud-based repository.

First, the institution needs to create a primary account that will govern the entire cloud-based infrastructure. The account does not have to be designated to a single individual since users of the cloud-based infrastructure will be addressed in the next section. To create an account, navigate to aws.amazon.com and create a primary account. Once the account has been created, it may take up to 24 hours for the account to access all the AWS services.

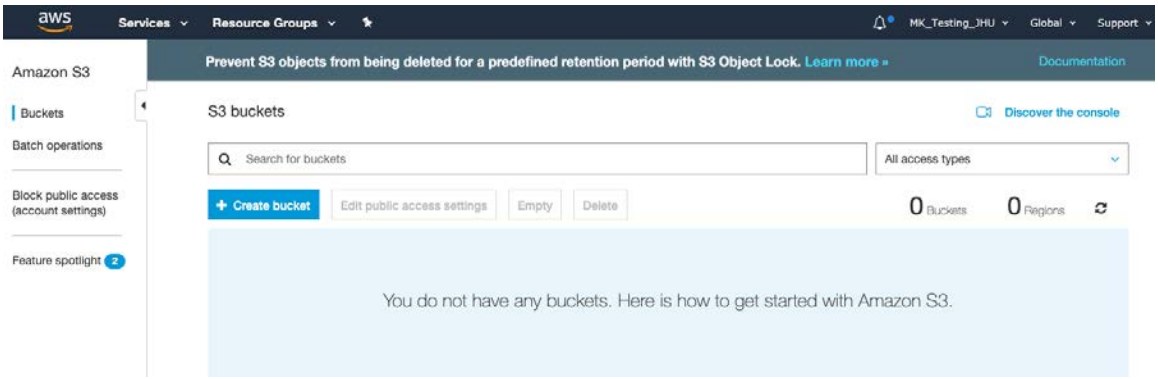
Once the account has been established, the user can navigate to the services tab on the screen. Under the Storage section, the user can select 'S3' as highlighted in Figure 1 below.

Figure 1: Services Tab[13]



Once the page navigates the user to the Amazon S3 page, the user can then proceed to create a bucket by clicking on the button shown in Figure 2.

Figure 2: Amazon S3 Main Page[14]



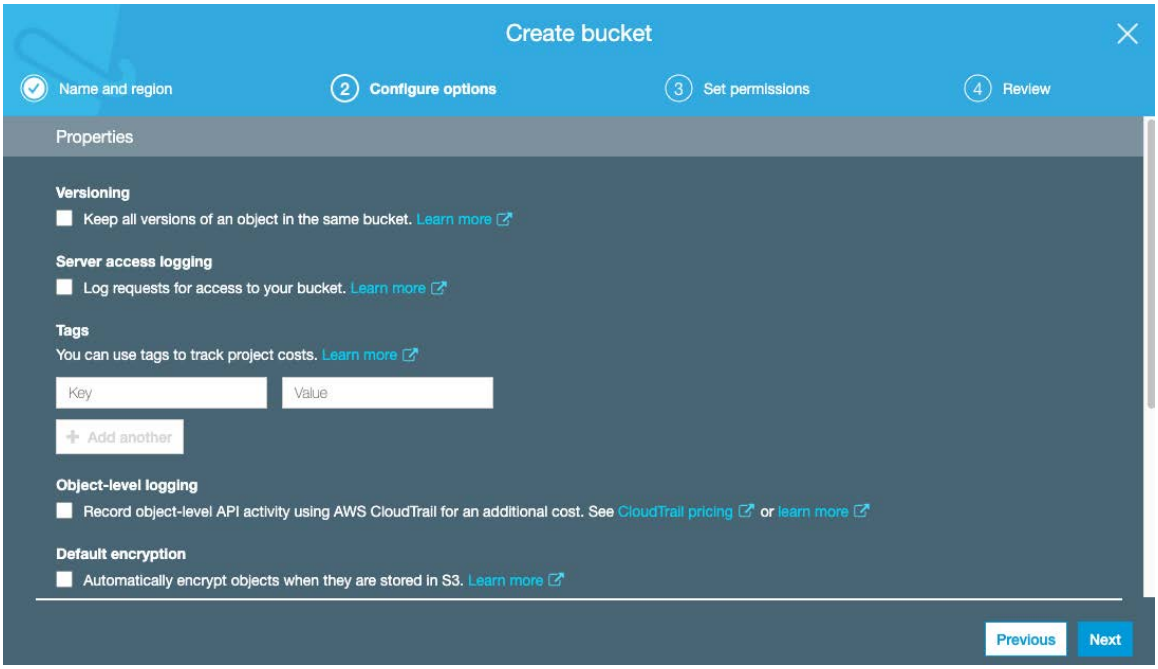
The bucket in Amazon S3 is another terminology for a folder that can be created on the desktop of a computer to store all the files. Buckets act as a folder in Amazon S3.

[13] Source Minsoo Kim, Figure 1: Services Tab
[14] Source Minsoo Kim, Figure 2: Amazon S3 Main Page

The S3 uses buckets because materials that are stored in the buckets are referred to as "objects." Oddly enough, these objects can be grouped and create a "folder." Therefore, the bucket is the folder that contains "folder," which contains "objects." However, the bucket can contain just the objects if they are not grouped into a folder.

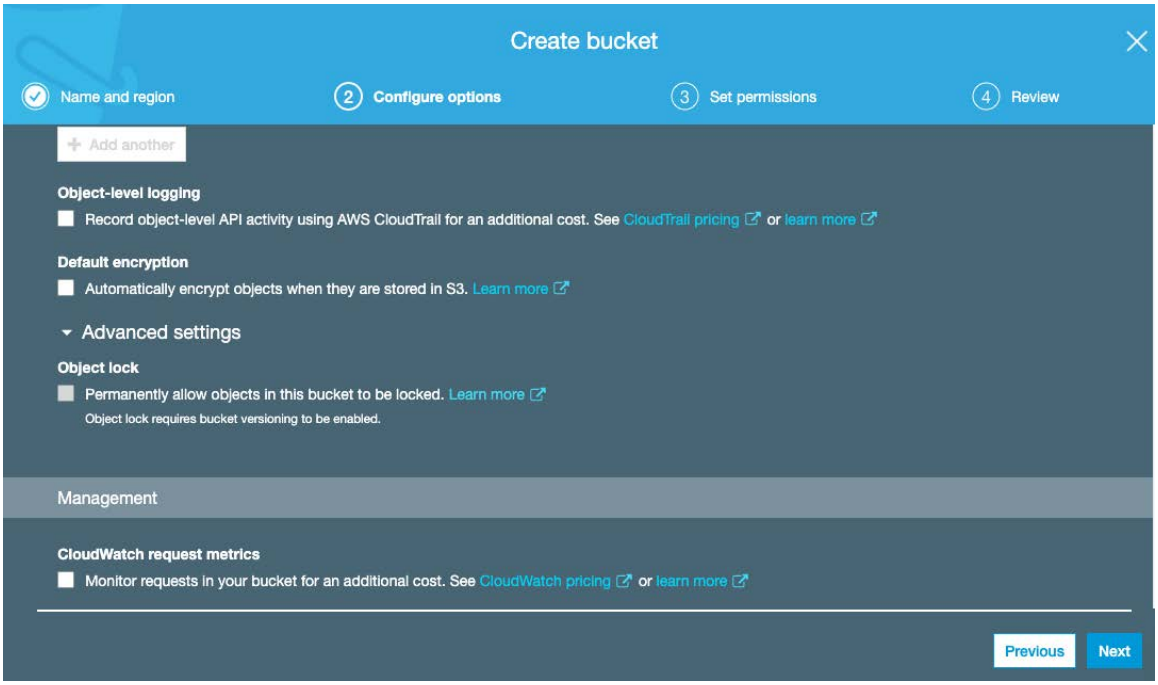
The bucket name, region, and settings from an existing bucket can be selected. Since this is the first bucket being created, the settings from an existing bucket option should be empty. However, the user can utilize the settings from the existing bucket instead of making the same settings for every bucket. Once the information has been filled out, click on the next button to proceed to further configurations. Next step, the bucket's configuration options can be selected. From Figures 3 and 4, the following properties are visible: versioning, server access logging, tags, object-level logging, default encryption, object lock, and CloudWatch request metrics.

Figure 3: Create Bucket Part I[15]



[15] Source, Minsoo Kim, Figure 3: Create Bucket Part I

Figure 4: Create Bucket Part II[16]



The versioning feature allows the user to keep all versions of the object uploaded in the bucket. Meaning, if the bucket went through any changes, the versions of the bucket before changes would be kept. All of the previous versions are kept and can be accessed by users. The server access logging allows the S3 to keep log requests for access to the bucket by all users. By turning on this feature, institutions can keep track of who last acquired access to the bucket. Tags are ways to label the buckets so that users can keep track of which bucket is. Object-level logging is a similar feature as the server access logging. The object-level logging keeps a tab on who tried to access the objects within the bucket. The default encryption feature refers to how the uploaded object would be encrypted while being stored in the S3 bucket. Under the advanced settings, the object lock feature "permanently allow objects in the bucket to be locked."[11] This feature

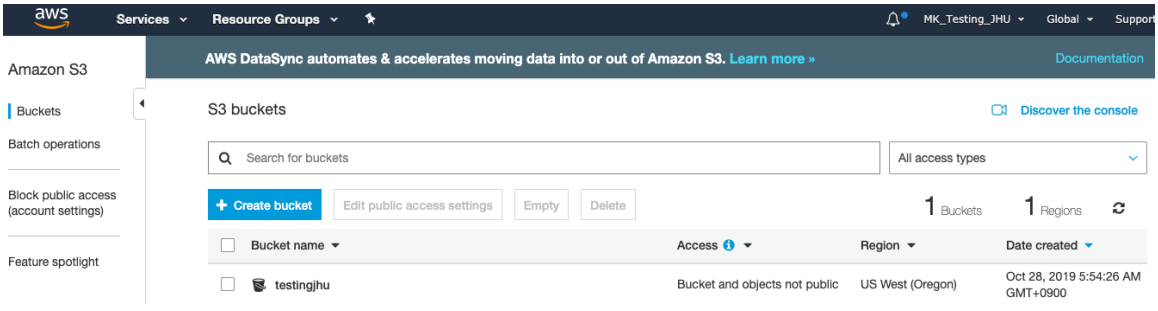
[16] Source, Minsoo Kim, Figure 4: Create Bucket Part II

prevents the "objects from being deleted to help ensure data integrity and regulatory compliance."[11] The object lock feature comes in handy to make sure that no user can accidentally tamper with the uploaded objects.

Lastly, the CloudWatch requests that metrics refer to another service, AWS CloudWatch. AWS CloudWatch monitors activities that happen within the bucket. If an unusual attempt to access the object in the bucket occurs, the administrator of the account will be notified of such an attempt. Not all features are free to use, but depending on the needs of the institution, these features can be utilized.

Once the configurations have been established, bucket permission settings need to be established before a bucket is created. The general advice is to block all public access, which denies external parties from accessing the bucket and its contents. For safety, the best practice is to avoid anyone from accessing the bucket. Instead, give permissions to the users who need to access the bucket. The setup for creating a bucket has been completed, and a bucket can be located in the Amazon S3 page, as seen in Figure 5.

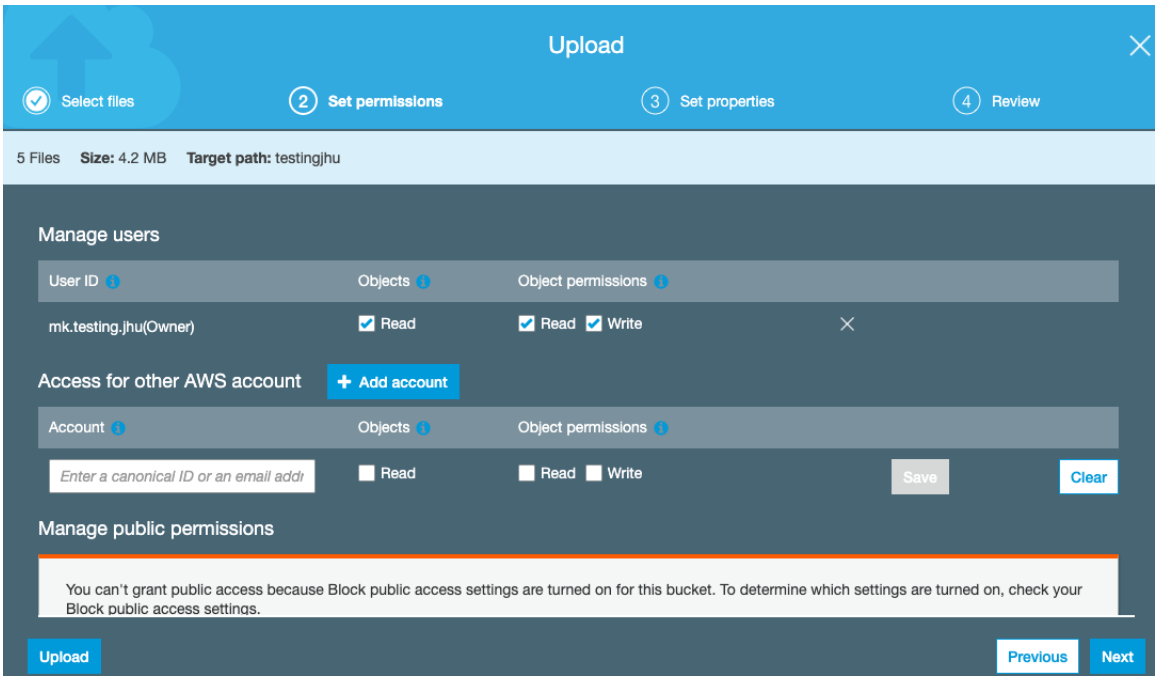
Figure 5: Successful Bucket Creation[17]



[17] Source, Minsoo Kim, Figure 5: Successful Bucket Creation

Now that the bucket has been created, documents can be uploaded as objects into the bucket. To upload, click on the upload button, and the user can add files to upload to the bucket. Multiple files can be selected and uploaded at the same time. Similar to the bucket creation process, objects require permission settings before the upload. Instead, shown in Figure 6, object permissions request the user to set access for other AWS account. Meaning that adding other users' AWS accounts allow those users to read or write the uploaded object.

Figure 6: Object Upload Settings[18]



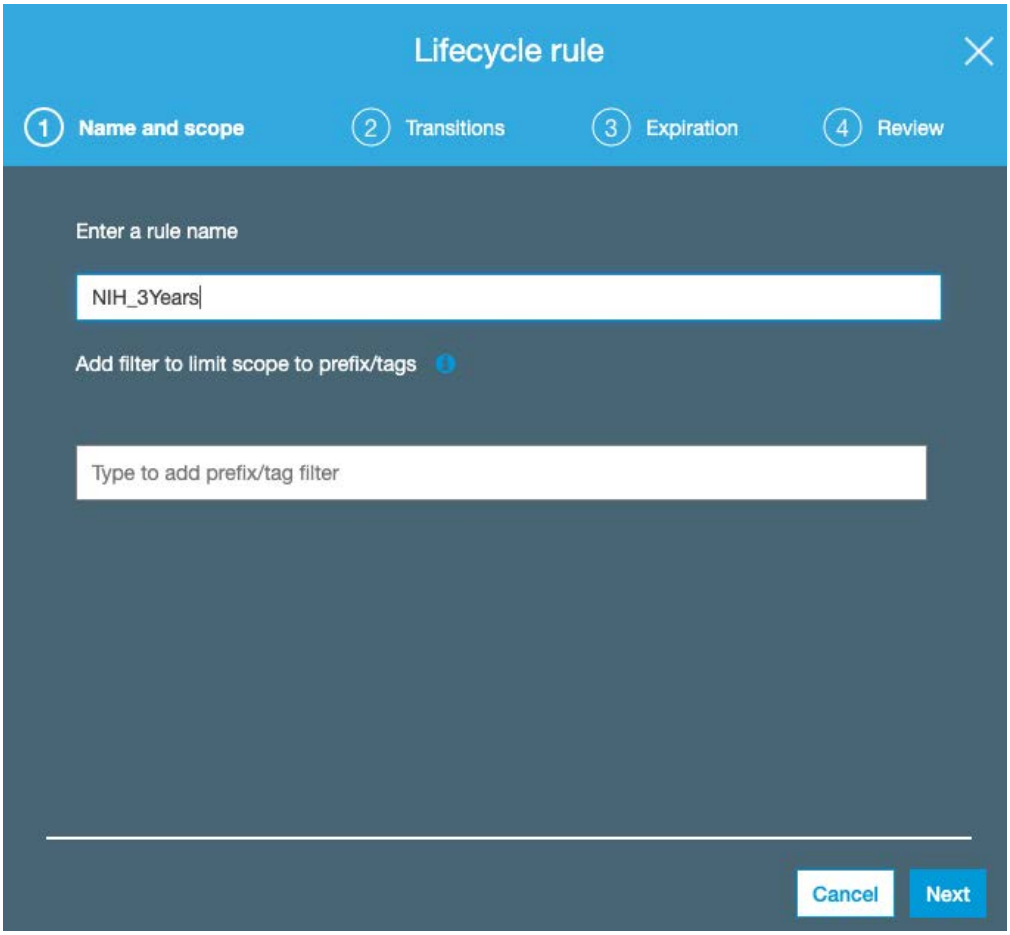
Lastly, the storage class needs to be established before uploading the objects into the bucket. Storage class refers to the tiers designed by the Amazon S3. Based on the use case of the uploaded objects and the frequency of the access, the appropriate storage class can be determined. It is essential to select the most appropriate storage class since the

[18] Source, Minsoo Kim, Figure 6: Object Upload Settings

storage class has different price rates. Once the storage class has been selected, the objects are ready to be uploaded in the bucket.

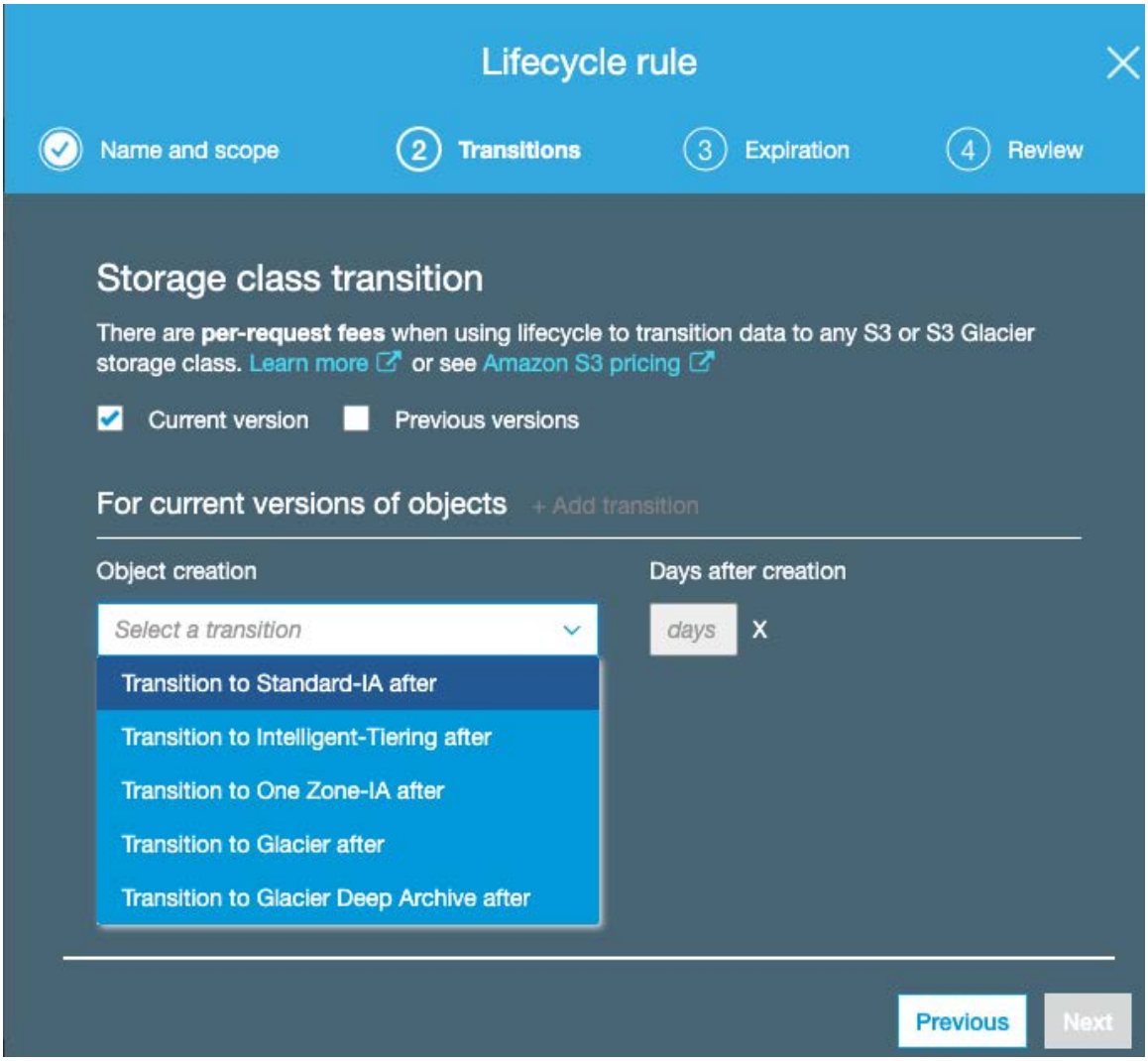
A cloud-based repository has been successfully created. However, there is an essential feature that needs to be addressed. Under the management tab in the bucket, the user can add a lifecycle rule to the bucket. A lifecycle refers to the period in which the bucket will exist in the Amazon S3 repository. Lifecycle policy can either terminate the bucket upon expiration or change the storage class of the objects in the bucket. By changing the storage class, the user can save more money by using a lower storage class tier. Setting a lifecycle policy rule is similar to that of the previous setup of uploading objects. Name the lifecycle rule, select storage class transition or configure expiration date, and complete the lifecycle rule setup. Figures 7, 8, and 9 illustrate the process of setting up a lifecycle rule on the Amazon S3 bucket.

Figure 7: Lifecycle Rule Part I[19]



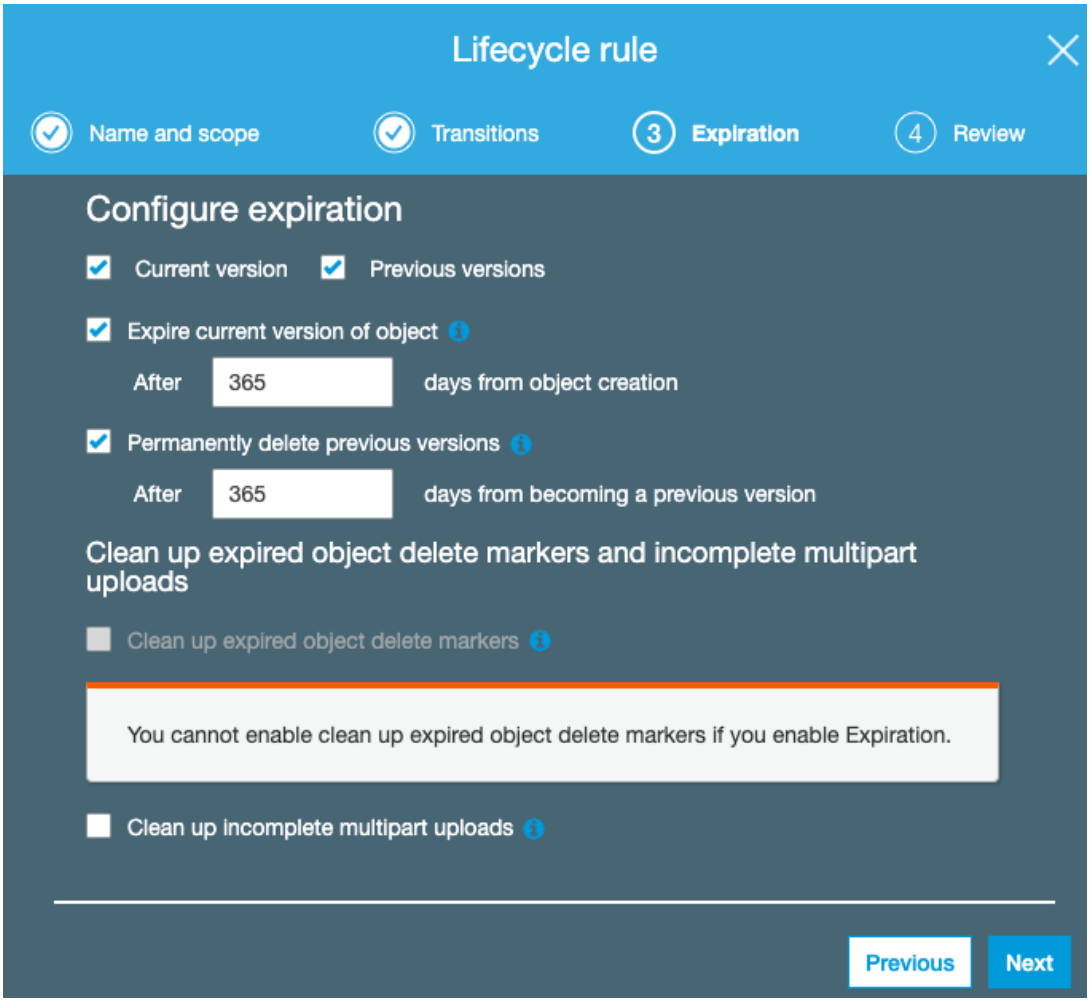
[19] Source, Minsoo Kim, Figure 7: Lifecycle Rule Part I

Figure 8: Lifecycle Rule Part II[20]



[20] Source, Minsoo Kim, Figure 8: Lifecycle Rule Part II

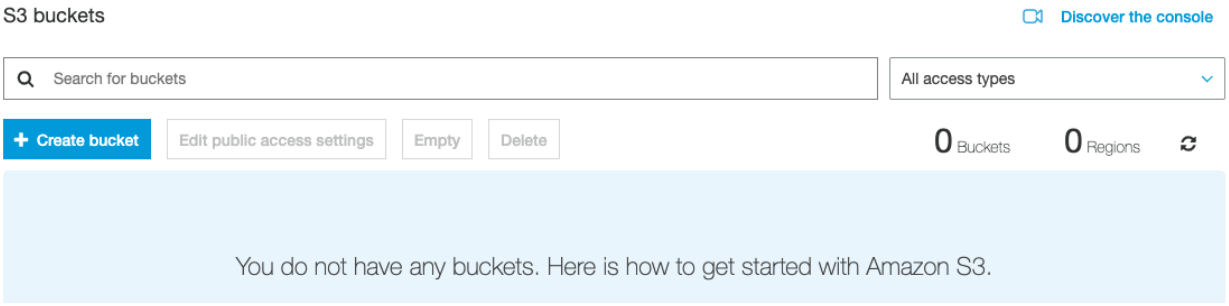
Figure 9: Lifecycle Rule Part III[21]



The following, Figure 10, depicts what the S3 main page would look like if the bucket were terminated by the lifecycle policy. AWS S3 would not contain any bucket since there was only one bucket in the repository and that bucket was terminated.

[21] Source, Minsoo Kim, Figure 9: Lifecycle Rule Part III

Figure 10: Lifecycle Policy in Effect[22]



With the added lifecycle policy, the institution can easily create a cloud-based repository to store documents that arose from research activity and terminate them upon expiration. Furthermore, if the user wanted to share the objects in the bucket, the user could allow access for users with AWS accounts.

6.2. Project Result 2. Managing User Access Guide

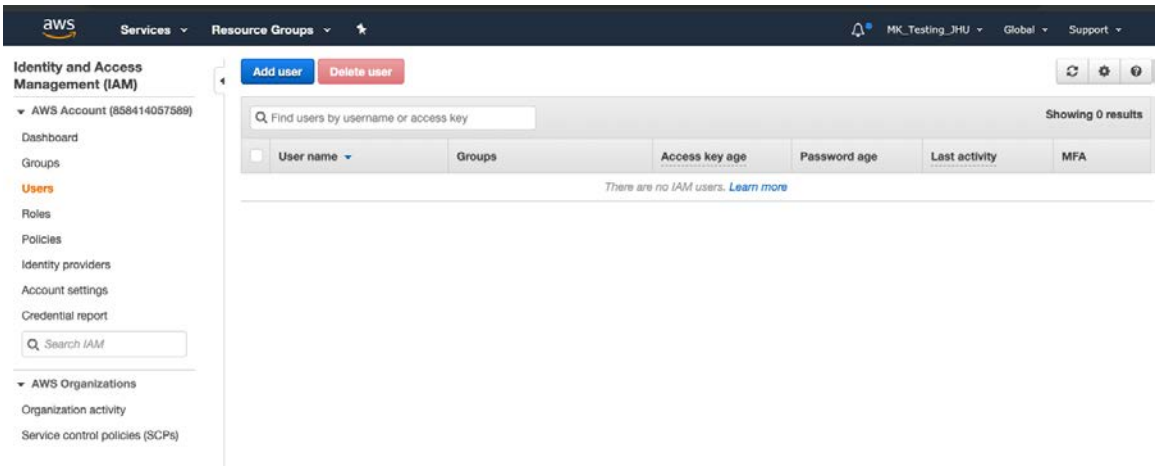
While the cloud-based repository has been created, the institution faculties cannot share a single account to access the repository. The research administrators should manage access to the cloud-based repository. To manage the user access of the cloud-based repository, the institution can utilize the service known as IAM Users. The Identity and Access Management (IAM) is a tool that allows the root administrator of the AWS account to grant or take away the permission of accessing services provided by AWS. By default, any new IAM user created has no access to any AWS services. Therefore, permissions, via access policy, needs to be provided by the administrator.

Managing user access is as simple as creating the cloud-based repository using Amazon S3. In the AWS Management Console Page, click on the "IAM" under the Security, Identity, & Compliance section. After being directed to the IAM page, click on

[22] Source, Minsoo Kim, Figure 10: Lifecycle Policy in Effect

the “Users” section on the left side of the panel. From there, click on the button, “Add User.” The following screen should resemble that of Figure 10.

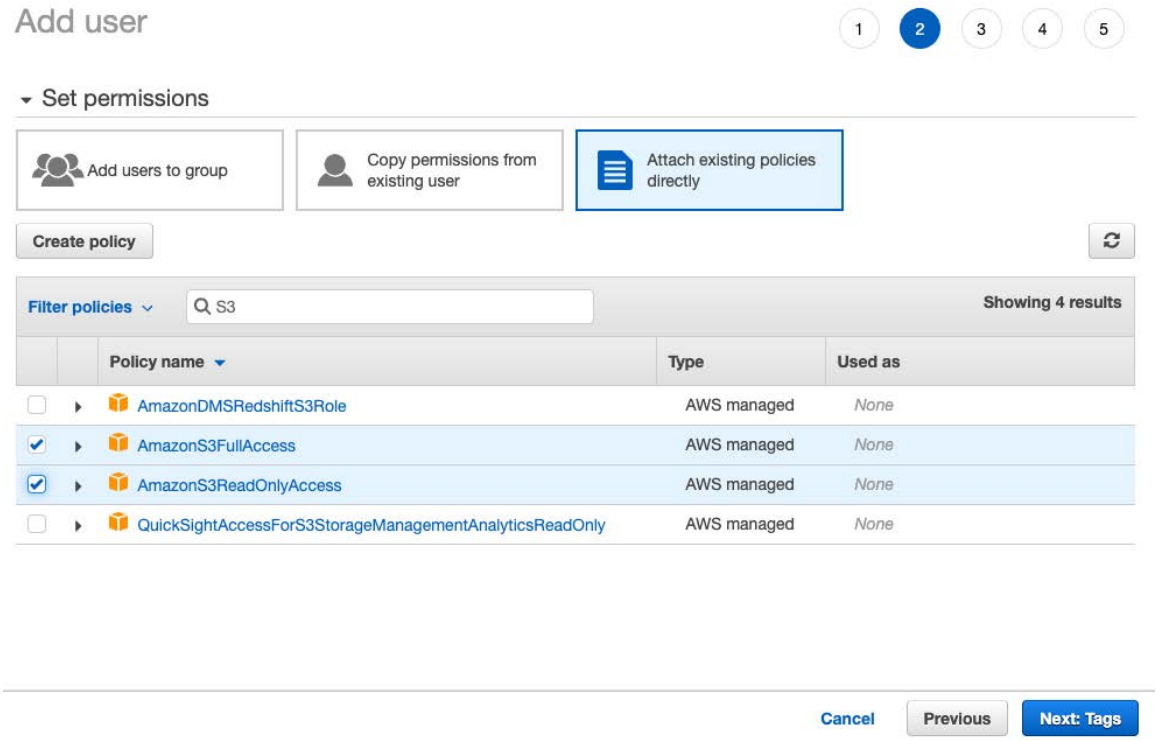
Figure 11: Add User Page[23]



Multiple users can be added and should be granted an AWS Management Console Access type. Console password can be autogenerated and have the user reset the password upon the first login. Next, users can be assigned to existing policies directly. As shown in Figure 11, type in “S3” and filter the policies to attach to the added users.

[23] Source, Minsoo Kim, Figure 11: Add User Page

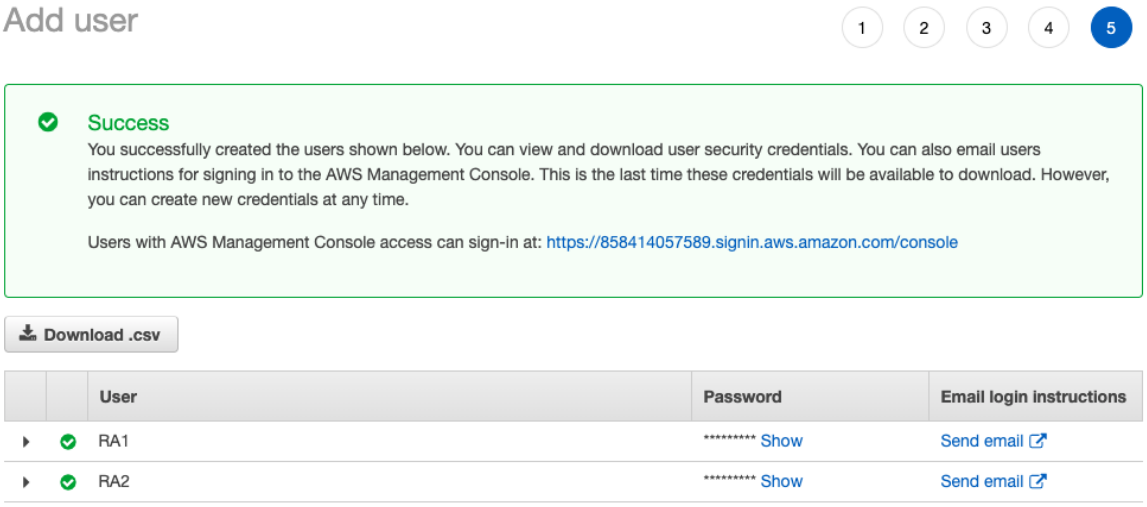
Figure 12: Access Policy[24]



Depending on the added users, full access or read-only access can be attached to the added users. Once the policy has been attached, review and the users are now successfully added to the IAM. On the success page, Figure 12, login instructions can be sent to the added users via email, and the added users can access the Amazon S3 per email instructions that they will receive.

[24] Source, Minsoo Kim, Figure 12: Access Policy

Figure 13: Success Page[25]



6.3. Project Result 3. Public Distribution Guide

Once the documents from the research activity have been successfully stored in the cloud-based repository, AWS S3, the documents may need to be accessed by non-faculty members. The retrieval of the documents may be requested due to various reasons: internal audits, research collaboration, and research integrity investigation. The cloud-based repository is created in a private environment – meaning that it is not visible to anyone unless permitted by the administrator. However, there are ways for administrators or users with access to share the uploaded objects with those who do not have access.

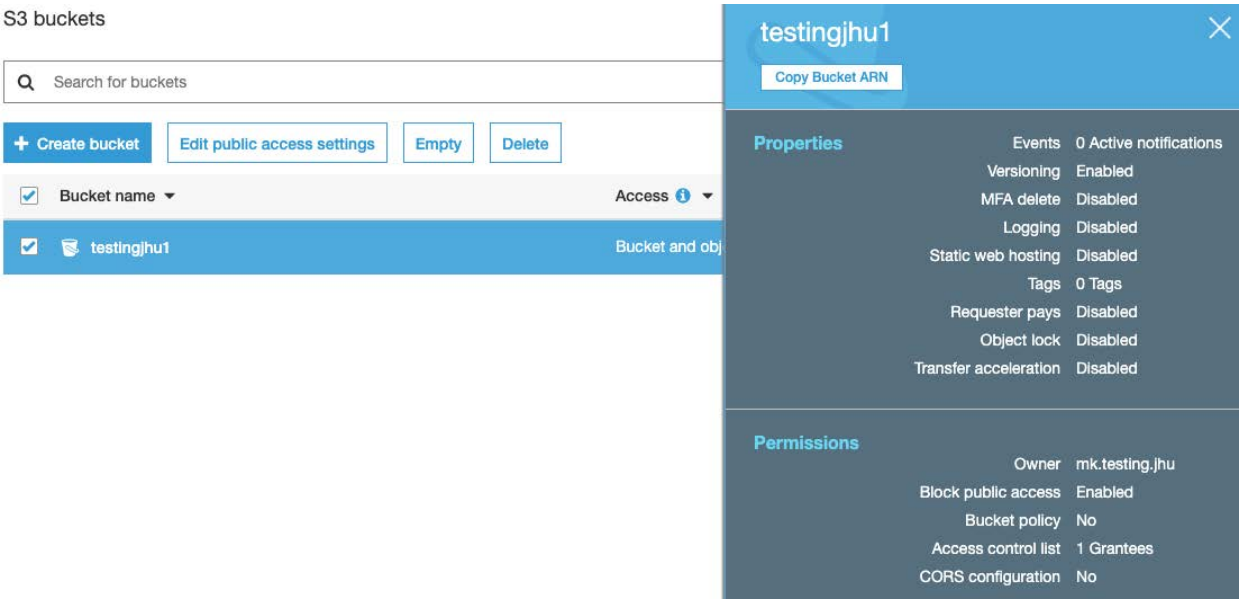
It is crucial to note that, without taking the following step, repository contents cannot be shared publicly with anyone without access to the repository. The bucket that contains objects must be made publicly accessible first on the bucket-level. If the bucket's public access setting is not granted, then the bucket and its contents cannot be

[25] Source, Minsoo Kim, Figure 13: Success Page

publicly available. Therefore, if the user has shared the bucket, but the outside recipient cannot see the contents, then the user must confirm that the bucket has been granted public access. It is important to note that by default, the bucket blocks public access upon creation. Therefore, the bucket's public access setting needs to be edited.

Shown on the following Figure 14, the user can see all the created buckets and their properties once the user selects the desired bucket. Under the permissions section, Figure 14 shows that the 'Block public access' has been enabled. Therefore, the bucket blocks all public access and prevents anyone from seeing its contents. If the bucket is made publicly available, the 'Block public access' property would state 'disabled.'

Figure 14: Bucket Properties[26]



To make the bucket available, click on the 'Edit public access settings' button next to the '+ Create bucket' tab. Once the user clicks on the settings tab, the user can

[26] Source, Minsoo Kim, Figure 14: Bucket Properties

uncheck the ‘Block *all* public access’ box, as seen in Figure 15. The user must save the adjusted settings to ensure the bucket can be shared publicly.

Figure 15: Bucket Public Access Settings[27]

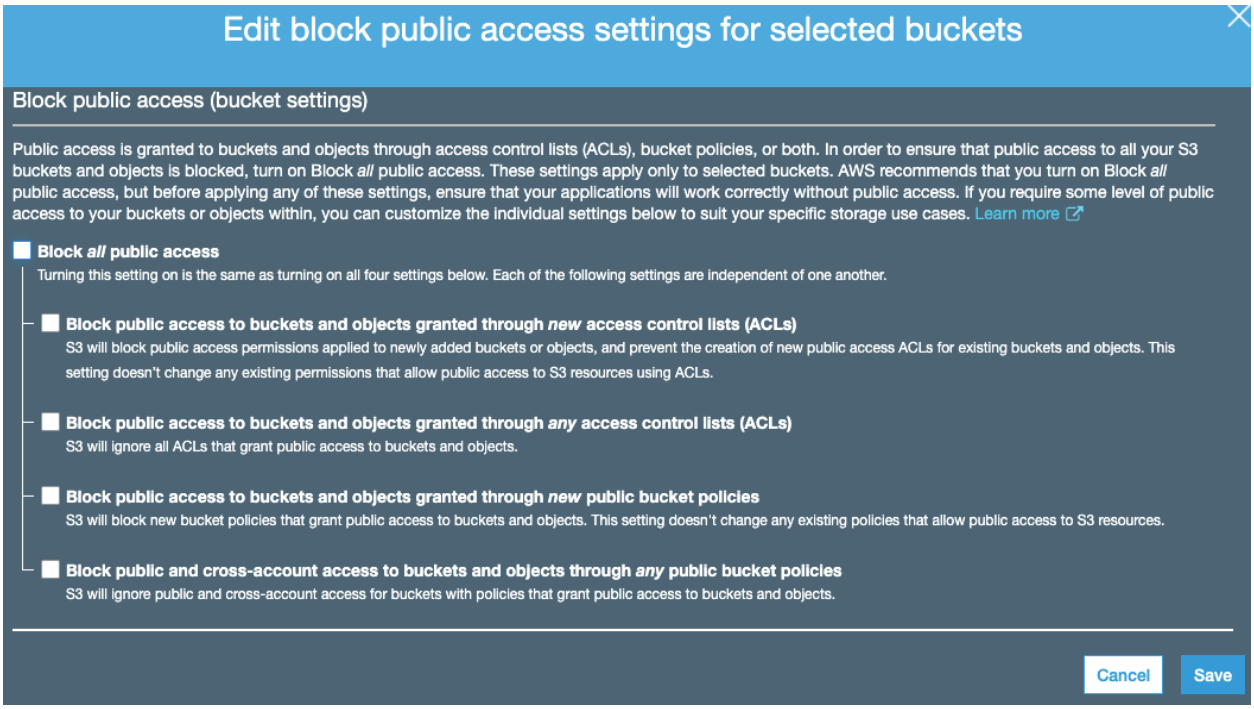
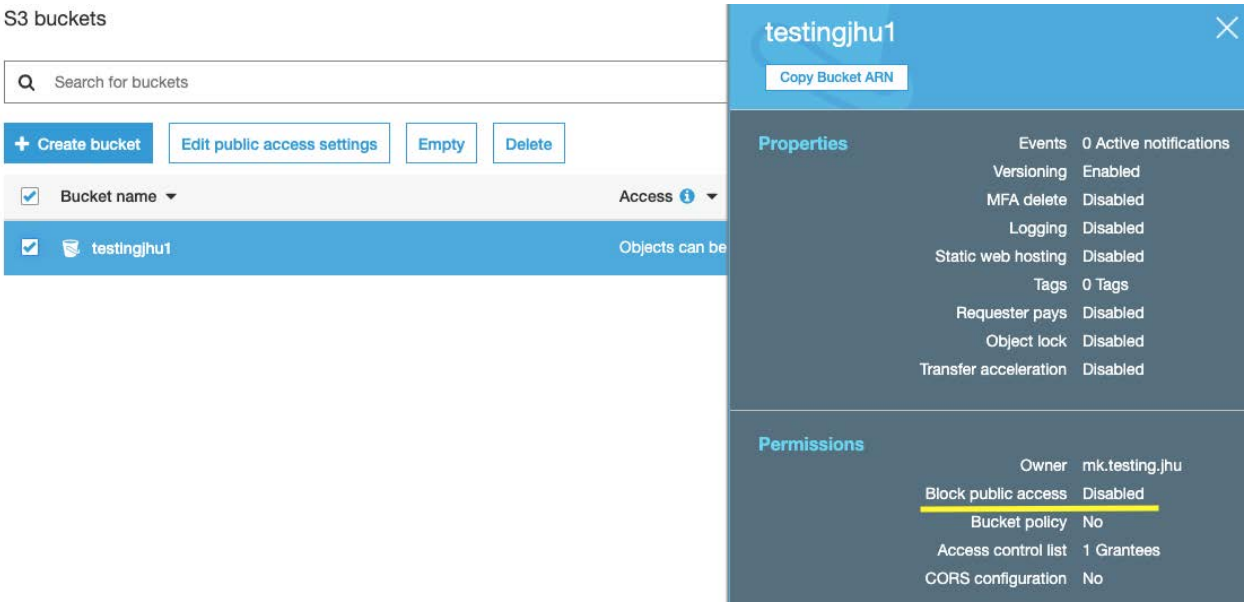


Figure 16 shows that the following changes made by the user are effective. Additionally, the user can note that the ‘block public access’ setting is disabled, highlighted by the yellow line in Figure 16.

[27] Source, Minsoo Kim, Figure 15: Bucket Public Access Settings

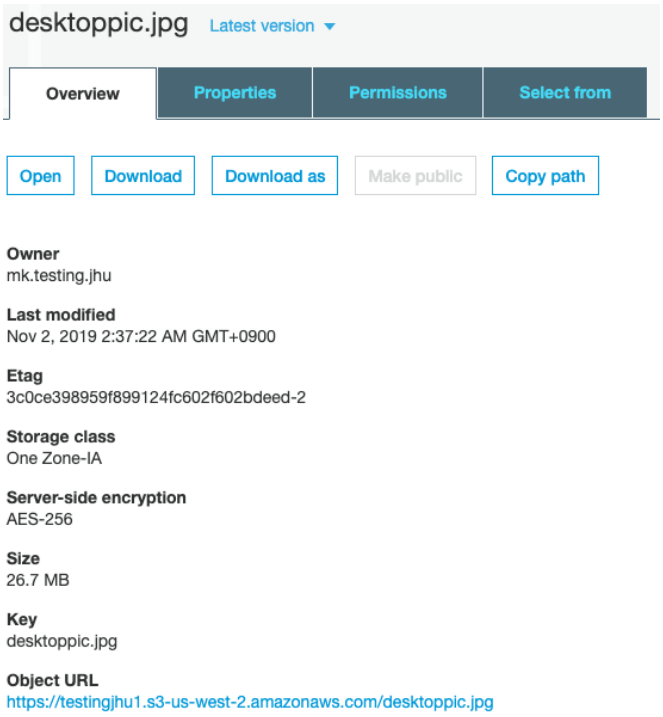
Figure 16: Confirmation of Bucket Public Access[28]



Now that the bucket grants public access, the user can select the objects inside the bucket to share with others. If the user clicks on the object, the user can see the property details of the object illustrated in Figure 17.

[28] Source, Minsoo Kim, Figure 16: Confirmation of Bucket Public Access

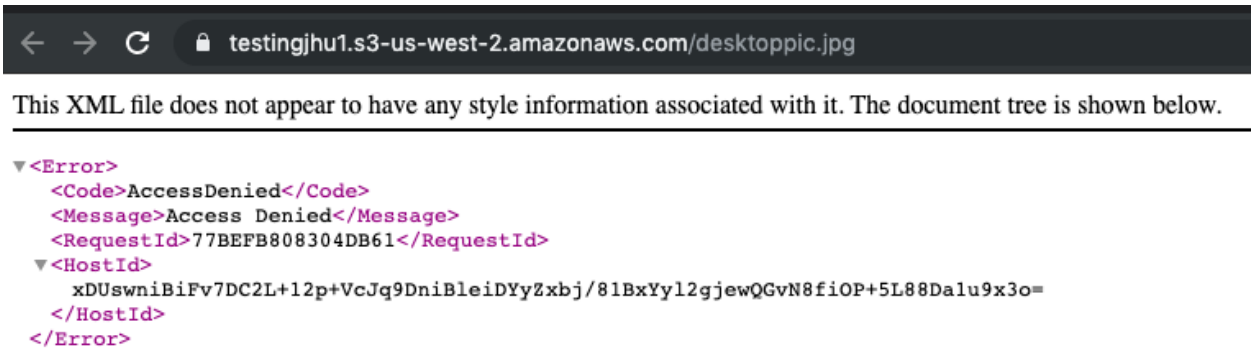
Figure 17: Object Properties Overview[29]



To share the object, the user needs to copy the object URL and send it to the designated recipient. The user can do the same for other objects that need to be shared with external parties. If the bucket were successfully made publicly available, the recipient would be able to view the contents of the bucket. However, if the bucket were not made publicly available, the recipient would see a screen resembling that of Figure 18.

Figure 18: Object Access Denied[30]

[29] Source, Minsoo Kim, Figure 17: Object Properties Overview
[30] Source, Minsoo Kim, Figure 18: Object Access Denied



Once the contents have been shared and accessed by recipients, the user needs to ensure that the bucket blocks public access again to protect contents in the bucket. If the bucket is not made private, the bucket becomes vulnerable to unwanted exposure to the public. Therefore, it is crucial for users that have access to the repository to ensure that buckets are made private once the sharing has been completed.

6.4. Project Result 4. Repository Usage Cost Management

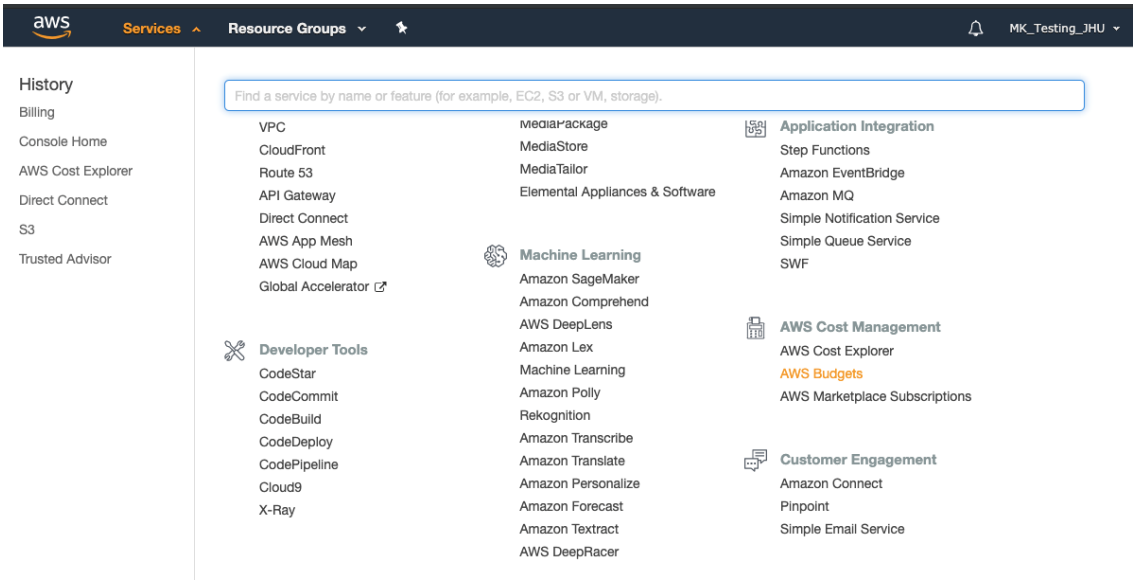
Using a cloud-based repository, institutions can track the financial statements for the usage of products and services in real-time. Furthermore, institutions can implement a threshold to the budget spent on the cloud-based repository. For the cloud-based repository created in this tutorial guide, there are two ways institutions can manage costs: AWS Budgets and AWS Cost Explorer.

AWS Budgets is a service offered by AWS, and it allows users to establish a budget in three different ways based on cost, usage, and reservation. For cost management in a cloud-based repository, reservation method can be neglected. If the institution wants to establish a total budget spent on the cloud-based repository, the institution can utilize the cost budget method to stay under that budget. If the institution wants to ensure that the users do not over-utilize the products and services in AWS and

create a huge bill, the institution can put a threshold in the usage budget method to prevent overspending by users.

To establish the budget, click on the ‘Services’ tab on the top of the main page after logging in to AWS. Under the ‘AWS Cost Management’ category, select the highlighted ‘AWS Budgets’ illustrated in Figure 19.

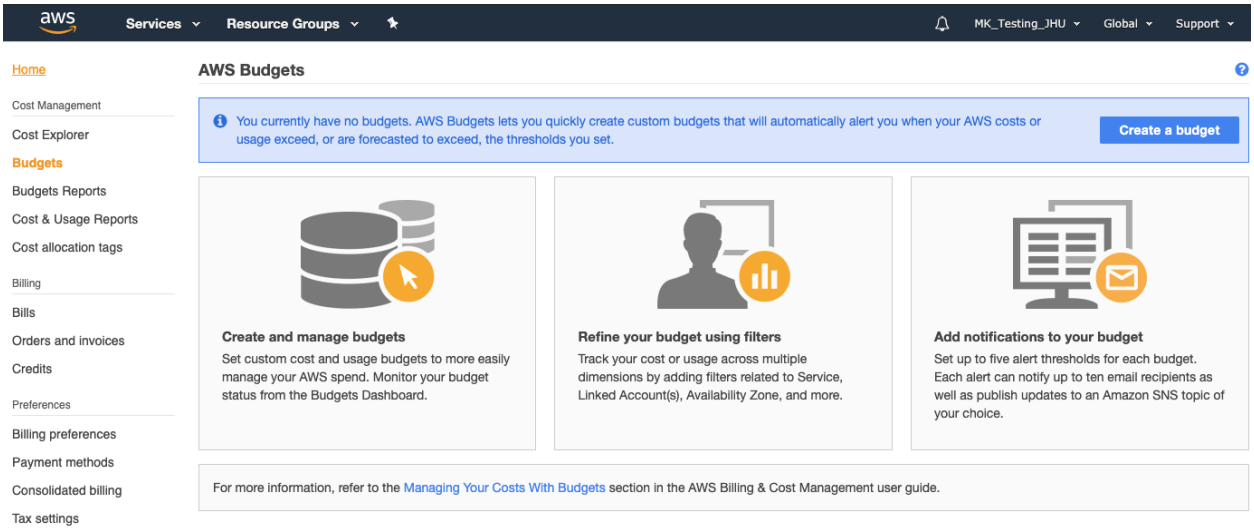
Figure 19: AWS Services Tab[31]



The user will be directed to the AWS Budgets page shown in Figure 20. To create a budget, click on the ‘create a budget’ tab highlighted in blue.

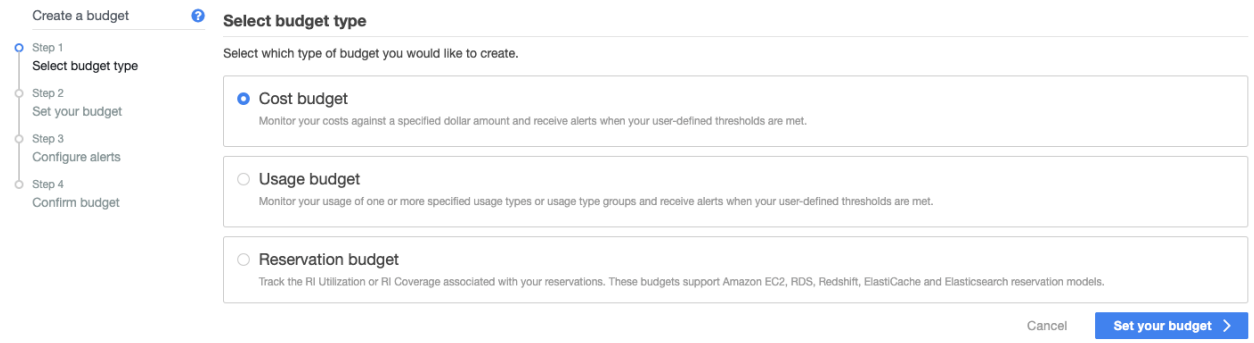
[31] Source, Minsoo Kim, Figure 19: AWS Services Tab

Figure 20: AWS Budgets Main Page[32]



The user can select to establish a budget from the three methods previously mentioned: cost, usage, and reservation. The cost budget method allows the user to put a total amount spent on the products and services in AWS. The usage budget method establishes a monetary threshold for a specific product and service in AWS. The three budget types are depicted in Figure 21.

Figure 21: AWS Budget Types[33]

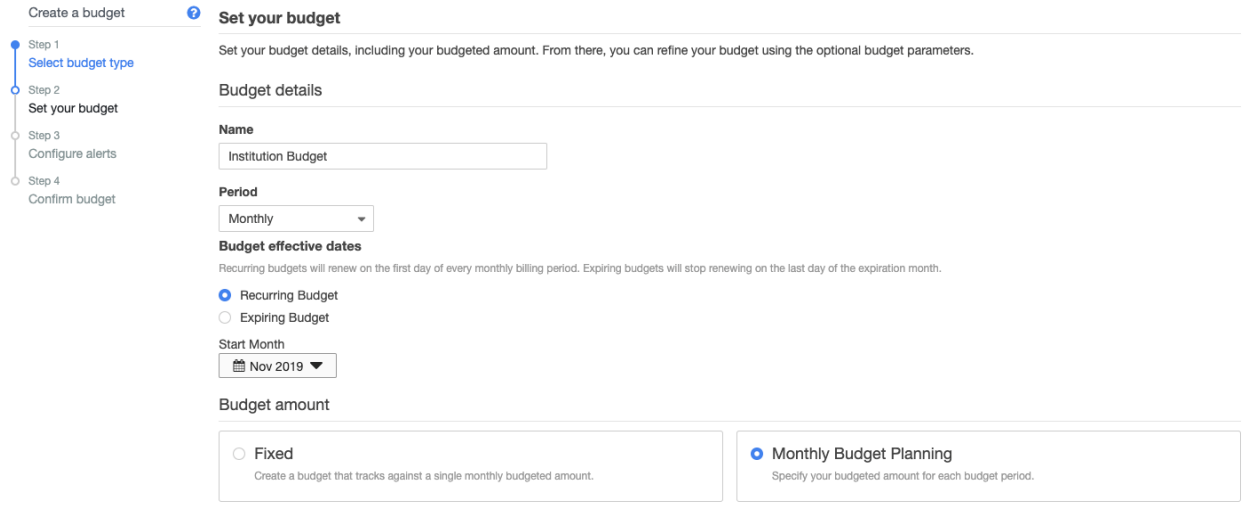


For the demo purpose, the tutorial guide selects a cost budget type method. Shown in Figure 22, the user needs to fill out the name of the budget, period, budget

[32] Source, Minsoo Kim, Figure 20: AWS Budgets Main Page
[33] Source, Minsoo Kim, Figure 21: AWS Budget Types

effective dates, and the budget amount. The user can name the budget to remember the type of budget that was established. The period section establishes how frequently – monthly, quarterly, and annually –the budget should occur. The budget effective dates allow users to implement a budget ahead of time. Furthermore, the user can recur a budget or expire a budget after one execution.

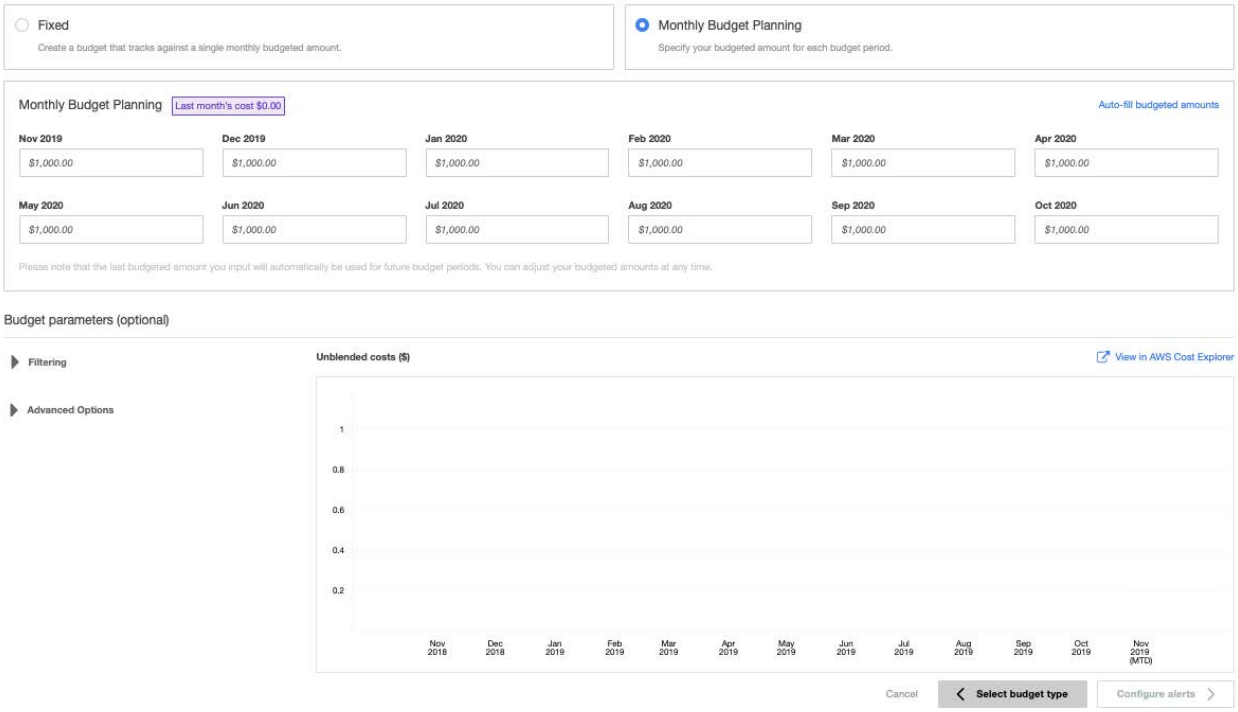
Figure 22: Budget Properties I[34]



Lastly, the budget amount can be established in two ways: fixed and monthly budget planning. Unlike a fixed budget, monthly budget planning lets the user decide if the budget can either increase or decrease over time. If the user selects the monthly budget planning option, the webpage allows the user to manually input the amount of monthly budget in the coming months. The layout of the monthly budget planning can be seen in Figure 23.

[34] Source, Minsoo Kim, Figure 22: Budget Properties I

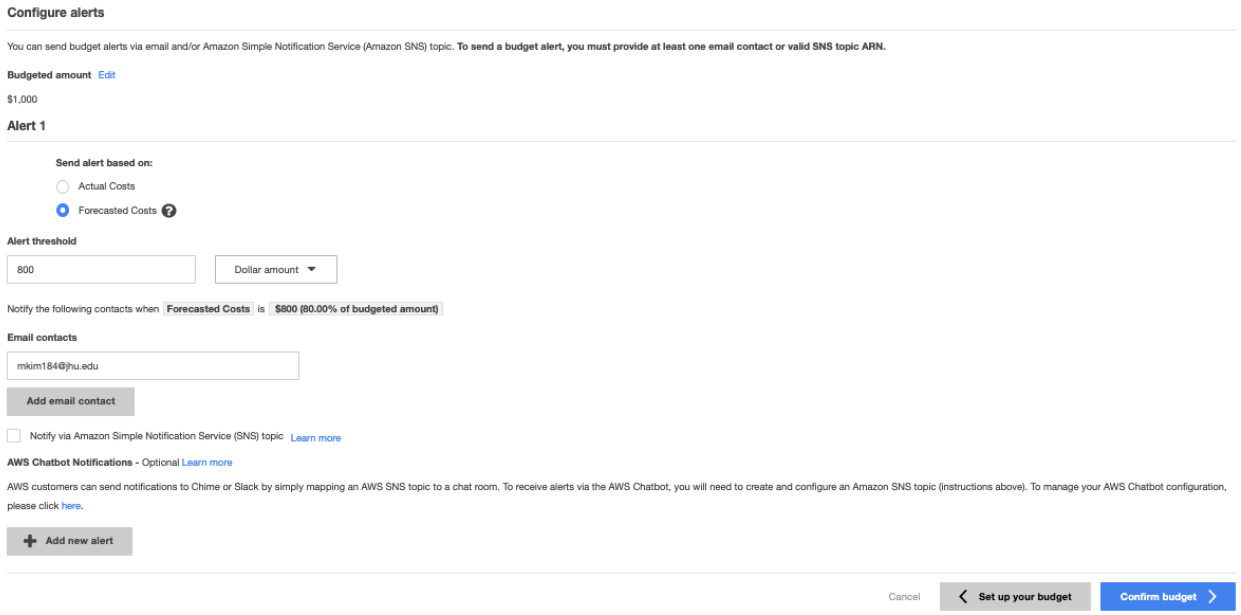
Figure 23: Budget Properties II[35]



Once the budget type has been created, the user must configure alerts in case the financial expenditure is getting close to the established budget. As seen in Figure 24, the user can configure alerts based on the actual costs or forecasted costs. The user can further establish the alert threshold to receive an alert if the cloud-based repository bill is too close to the threshold. Lastly, the user can add email contacts that can receive alerts and establish more than one alert for the budget.

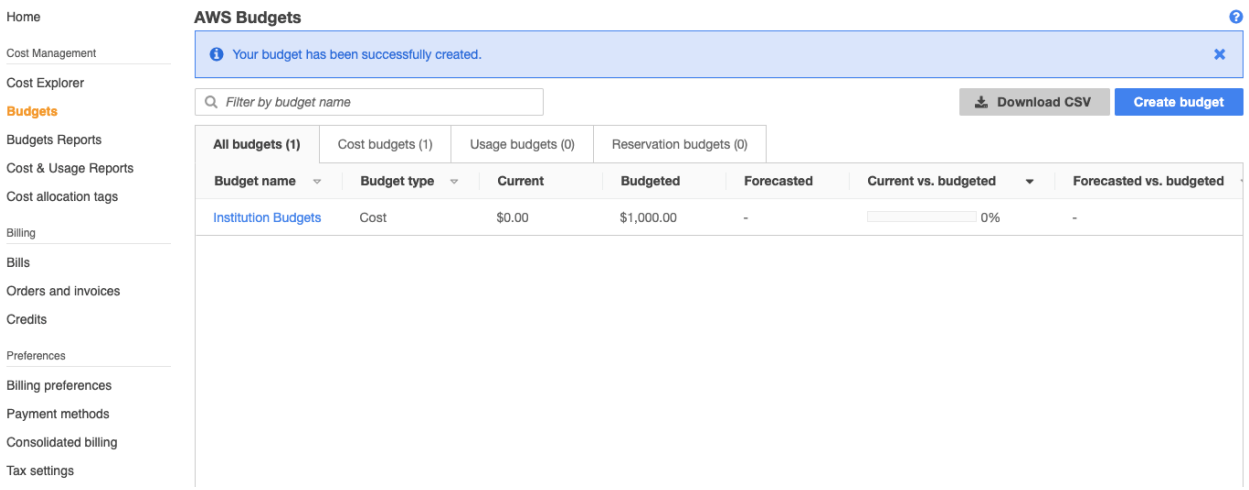
[35] Source, Minsoo Kim, Figure 23: Budget Properties II

Figure 24: AWS Budget Alert Configuration[36]



Once the settings have been filled out, the user can select 'confirm budget' and create a budget. Noted in Figure 25, the user has successfully created a cost budget, named Institution Budgets, that has a threshold of $1,000.

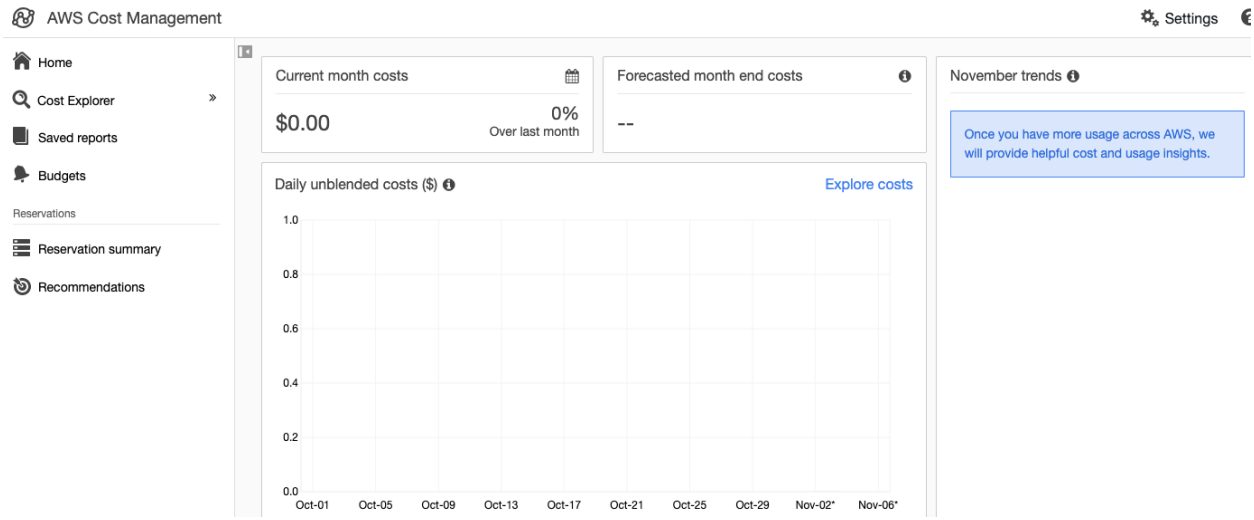
Figure 25: Successful AWS Budget[37]



[36] Source, Minsoo Kim, Figure 24: AWS Budget Alert Configuration
[37] Source, Minsoo Kim, Figure 25: Successful AWS Budget

Alternatively, AWS Cost Explorer provides an overview of the total usage of AWS products and services. The significant feature of the AWS Cost Explorer is the predicted cost for the next month. AWS Cost Explorer presents a predicted value of how much the user may utilize in the next month based on the past three months of data. As depicted in Figure 26, the user can see the current month costs, forecasted month end costs, and the chart of daily costs. Figure 26 is for a demonstration-purpose, so it does not show actual numbers spent on the cloud-based repository.

Figure 26: AWS Cost Management[38]



[38] Source, Minsoo Kim, Figure 26: AWS Cost Management

Chapter 7. Recommendations and Discussion

7.1. Introduction

The literature review led to the understanding that institutions are implementing a cloud-based infrastructure in efforts to reduce administrative burdens. The administrative burden is significant enough to impact faculty from efficiently carrying out the research. Therefore, institutions have sought out a way to automate, reduce, and simplify administrative tasks to stop the effort in research from being diverted. Institutions have been implementing commercial software-based solutions to adapt a cloud-based infrastructure to address administrative burden in three parts: pre-award, award, and post-award. Based on different needs, institutions use products and services provided by different vendors.

The guide in this project allows institutions that do not have the cloud-based infrastructure to adopt a cloud-based repository to handle record management in the post-award phase. However, the guide is general and limited to a repository segment within the post-award phase. Therefore, the following recommendations are made for institutions to make use of to have a smooth transition into a cloud-based infrastructure.

7.2. Recommendations

7.2.1. Recommendation 1. Use Both Cloud-based Infrastructure and Current System Implemented for the Institution.

While the benefits of using a cloud-based infrastructure may be evident, not everyone keeps up with the change in the system immediately. Additionally, stakeholders can potentially disagree with the necessity to migrate from the current system to a cloud-based infrastructure in order to handle the administrative burden. These are both feasible

possibilities that could occur at any institution looking to make a transition into a cloud-based system. For such a case, it is recommended that the institution utilize both the current system and a newly adopted cloud-based infrastructure as a hybrid model.

Despite the FDP study, which indicates the administrative burden to be severe, some faculties may not agree with the study. It is possible that the faculties have already adapted to the current system, and it is more difficult for them to try to get used to a newly implemented system. The benefit of adopting a cloud-based infrastructure is the flexibility of using a hybrid model. Ultimately, the institution can use some parts of the current system and handle the rest of the workload with the cloud-based system. The project provides a guide to creating a cloud-based repository for the post-award phase; thus, the institution can continue to use the current system for the pre-award and award phases. Institutions can slowly adopt a cloud-based infrastructure to handle both pre-award and award phases in the future.

7.2.2. Recommendation 2. Customize the Cloud-based Infrastructure to Handle the Complete Award lifecycle.

Institutions use products and services from different vendors that suit their needs in each phase of the award lifecycle. In this project, the guide focuses specifically on creating a cloud-based repository to handle record management in the post-award phase. However, the guide is created on an AWS cloud platform, which provides diverse services to handle the complete award lifecycle. Instead of using multiple software-based solutions to migrate to a cloud-based infrastructure, institutions could focus on utilizing a single platform that can provide needed products and services. AWS is a customizable

platform, so it is possible to create a universal platform for all institutions of higher education across the US.

The following is an example of the tasks that can be accomplished using services by AWS to handle pre-award and the award phases:

1. Use Elastic Compute Cloud (EC2) to create a proposal submission system. The system can be web-based and require the faculties to fill-out and submit proposals through the system. Furthermore, the EC2 can be used to create a search engine that can navigate through funding opportunities posted in the federal agencies.
2. Once the proposal has been submitted, Simple Work Flow (SWF) can be administered to track the progress of the proposal submission. SWF is a service that administers the entire workflow of a process. Thus, the SWF can be used to track the progress of the proposal and manage the budget usage of international collaboration.
3. Amazon Simple Notification Service (SNS) can be set up so that the faculty receives notification of the progress or alerts in real-time. Additionally, SNS can be used to inform the faculty of the budget expenditure. If the budget expenditure of an award is reaching 80% of the threshold, the SNS can be triggered to alert the faculty. Alternatively, the SNS can be triggered to let faculty know that proposal submission requires additional documents.

The example illustrates diverse and sophisticated services that AWS provides for institutions to take advantage of and improve the adopted cloud-based repository to handle the complete award lifecycle.

Chapter 8. Conclusion

The Federal Demonstration Project discovered that administrative burdens are very significant and bring impacts to the productivity of research. FDP questionnaire on administrative burden issued in 2009 reported that researchers spent 42% of their time spent on the research is spent on research-related administrative tasks.[39] To reduce the administrative burden and increase efficiency in research, institutions have sought to implement a new solution: a cloud-based infrastructure. Institutions are adopting cloud-based infrastructure to reduce administrative burdens and automate repetitive tasks.

Cloud-based infrastructures are implemented by using commercial software-based products and services. These products and services provide a cloud-based infrastructure that handles three segments of the award lifecycle: pre-award, award, and post-award cycle. There are numerous vendors in the market that institutions apply the best option that is suited for them. However, these services on the market are not compatible with each other. The lack of compatibility makes it difficult for institutions to share the information with other institutions retained by the cloud-based infrastructure. Therefore, institutions should seek to find a platform that can provide products and services that can handle all their needs. Amazon Web Services (AWS) is a cloud platform that can adopt various needs by institutions across the country, and potentially bring a universal cloud-based infrastructure to handle the award lifecycle.

Institutions can relieve administrative burdens and ensure the effort in research are expended in performing the research rather than being diverted in related

[39] Sara Rockwell. "The FDP Faculty Burden Survey." Research management review. U.S. National Library of Medicine, 2009. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2887040/.

administrative tasks. Nevertheless, many institutions have not implemented a cloud-based infrastructure and still struggle with these issues. Therefore, this project has provided a guide for institutions to utilize and adopt a cloud-based repository, specifically for the post-award phase. Therefore, the guide is not too complicated for research administrators to follow, yet sophisticated enough to handle the basic needs of the institution. The guide is not designed specifically for a targeted institution, but it is designed to be malleable. Institutions have different systems and needs, so they need a cloud platform that can adapt and handle the required needs in all phases of the award lifecycle.

The author hopes that by utilizing this guide, institutions can effortlessly try out a cloud-based repository and implement a new solution to address concerns raised by administrative burdens. Furthermore, the guide seeks to provide institutions general instructions to customize and improve the adapted cloud-based repository so that the cloud-based infrastructure can address all stages of the award lifecycle.

Bibliography

"Amazon S3 Pricing," Last modified 2002. https://aws.amazon.com/s3/pricing/?nc=sn&loc=4.

Perlitz, Lee and Elliott, Steven G. "The Products." Amazon. Pearson Education Australia, 2000. https://aws.amazon.com/products/storage/?nc2=h_ql_prod_st.

"PeopleSoft Enterprise Grants Management." Oracle. Accessed October 25, 2019. http://www.oracle.com/us/products/applications/peoplesoft-enterprise/service-automation/peoplesoft-grants-management-065800.html.

"PeopleSoft Grants." PeopleSoft Grants - University of Houston, Last modified September 26, 2019. https://www.uh.edu/research/sponsored-projects/peoplesoft/.

"Post-Award Management Software: Monitor and Report Faster." Cayuse. Accessed October 22, 2019. https://cayuse.com/post-award/.

"Research Administration System." The Research Administration System | Controller's Office. Accessed October 25, 2019. https://controller.ucsf.edu/how-to-guides/contracts-grants-accounting/research-administration-system.

Rockwell, Sara. "The FDP Faculty Burden Survey." Research management review. U.S. National Library of Medicine, 2009. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2887040/.

Saas, Tyler Kemp, James Deloitte Consulting LLP, It Takes an Eco-System: A review of the Research Administration Landscape, Research Management Review, Volume 22, Number 1 (2017)

"University of San Diego Case Study." Cayuse. Accessed October 25, 2019. https://cayuse.com/case-study/university-san-diego-case-study/.

Appendix 1: Tutorial Guide for Institutions of Higher Education to Create a Cloud-based Repository Using Amazon Web Services (AWS)

Tutorial Guide for Institutions of Higher Education to Create a Cloud-based Repository Using Amazon Web Services (AWS)

by
Minsoo Kim

Introduction

The current practice of research administration requires a substantial amount of responsibilities from the research administrators. Research administrators are invested in not only assisting institution's faculty members in obtaining funding for research, but they are also devoted to ensuring that institutions are compliant with regulations that come with the funding. The competition for receiving funding has increased over the years. Additionally, the needs demanded by the sponsors are shifting over time. Institutions are compliant with varying regulations by different sponsors to ensure their relationships are in good standing. However, research administrators are burdened by an abundance of tasks that need to be fulfilled to maintain such relationships. The effort to reduce administrative burdens for research administration has become vital in the efficiency of institutions.

To address the administrative burdens, institutions seek solutions that can automate tasks for research administrators. One of the solutions includes the automation of record management in the post-award phase. Research data and records that arose from research activity are required to be kept for a certain amount of period and discarded upon expiration. Institutions have sought to implement software solutions to automate the process of record retention and deletion. Therefore, this handbook guides institutions and other entities to create a cloud-based repository and migrate their current physical record management systems into a software-based solution. The handbook solely uses services and products provided by Amazon Web Services (AWS).

Table of Contents

Chapter 1. Why a Cloud-based Repository?

1.1. Purpose

Upon closing out an award, researchers are required to retain the records from the research activity for a certain amount of period. The period required for retention varies depending on the sponsor's requirements. Once the period for recordkeeping has expired, documents from the sponsored projects must be discarded. The award cycle comes to an end when the documents from the research activity are discarded upon expiration. Institutions have implemented various record management services suited to their needs. The record management services, under the scope of research administration, requires physical space to retain documents and discard them when expired.

These records need to be destroyed when expired to comply with various regulations. Records retained during the required period of time are subject to audit. Records retained beyond the record retention time period, if available, are subject to recall when there is an audit or internal investigation. Records destroyed in a timely manner in accordance with federal, state and local regulations are not available for audits and investigations. Thus, the IHE is at a reduced risk of audit findings if the records are destroyed at the appropriate time. Such responsibility for record destruction can be automated, utilizing a software-based infrastructure to store the documents away and discard them when they are expired. Instead of increasing the workload of research administrators, utilizing a software-based cloud repository could quickly alleviate such tasks.

1.2. Significance

This handbook is significant for institutions, especially institutions of higher education (IHE), to increase the efficiency of research administrators and to reduce burdens to store and destroy documents. Explicitly, the guide incorporates AWS S3 to create the repository. AWS S3 is a service known to provide the most cost-efficient and unlimited amounts of storage. Institutions will never have to worry about running out of storage capacity of the repository created with AWS S3. Furthermore, utilizing less physical storage capacity can reduce the Indirect (F&A) costs. Lowered F&A costs would allow researchers to utilize more of the funding received from the sponsors. Furthermore, when institutions expand facilities, institutions can focus on utilizing capacities for purposes other than physically storing documents. In addition, the human error of losing the stored documents can be reduced even further by abiding by the guidance created from the project. Ultimately, the cloud-based system for record management system could be an initial step towards creating a universal platform for institutions to share research data in encryption.

1.3. Exclusions and Limitations

The handbook is intended to demonstrate using the cloud-based infrastructure repository as an excellent alternative to physical record management system in the research administration. Therefore, the handbook provides tutorial on how to create a cloud-based infrastructure. This guide will work as general guidance and is not intended

for a specific institution. There are other software solutions that institutions can utilize to create a cloud-based infrastructure other than the AWS. The estimated usage cost is an example; thus, the actual costs of using the cloud-based repository may differ per use case. It is challenging to create a universal comparison between the current institutional structure and cloud-based repository since institutions differ from one another.

Chapter 2. Creating a Cloud-based Repository Using AWS S3

2.1. Getting Started

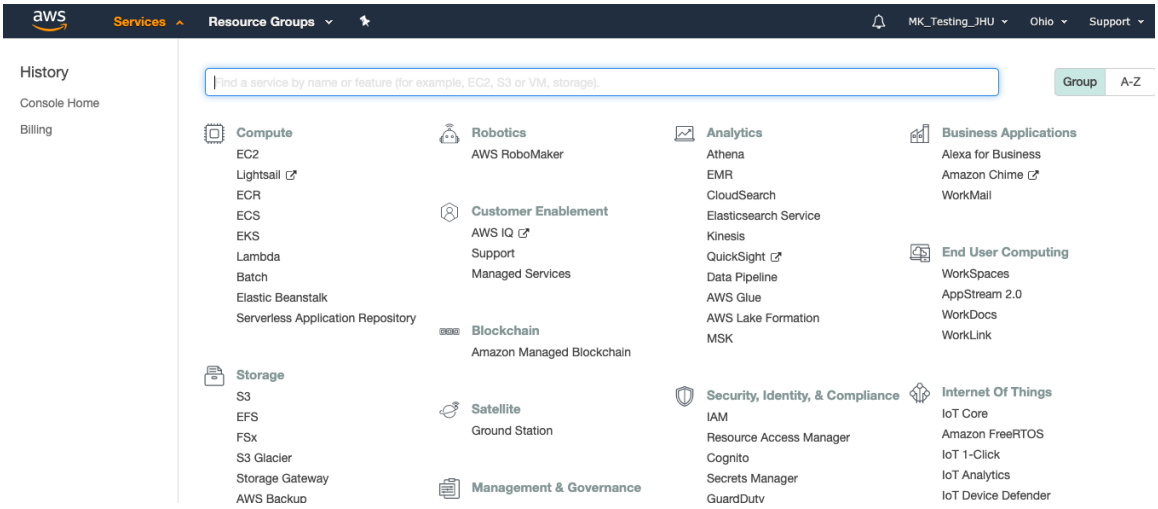
The outcome of this handbook is a tutorial guide for institutions to create a cloud-based repository using the services and products by the AWS. The guide is not targeted for a specific institution, but it is designed to be adapted by many institutions that do not have a cloud-based infrastructure. The guide can be used as a trial for institutions before fully migrating over to a cloud-based infrastructure. Overall, the tutorial illustrates how to set up the services to create a cloud-based repository, manage access to users that will utilize the repository, distribute the uploaded documents in the repository, and financially manage the repository.

The Amazon Web Services (AWS) is a cloud platform, most widely used, that provides customizable solutions and products. Any individuals and entities can easily use the services. Therefore, the guide utilizes the AWS cloud platform to create a cloud-based repository.

First, the institution needs to create a primary account that will govern the entire cloud-based infrastructure. The account does not have to be designated to a single individual since users of the cloud-based infrastructure will be addressed in the next section. To create an account, navigate to aws.amazon.com and create a primary account. Once the account has been created, it may take up to 24 hours for the account to access all the AWS services.

Once the account has been established, the user can navigate to the services tab on the screen. Under the Storage section, the user can select 'S3' as highlighted in Figure 1 below.

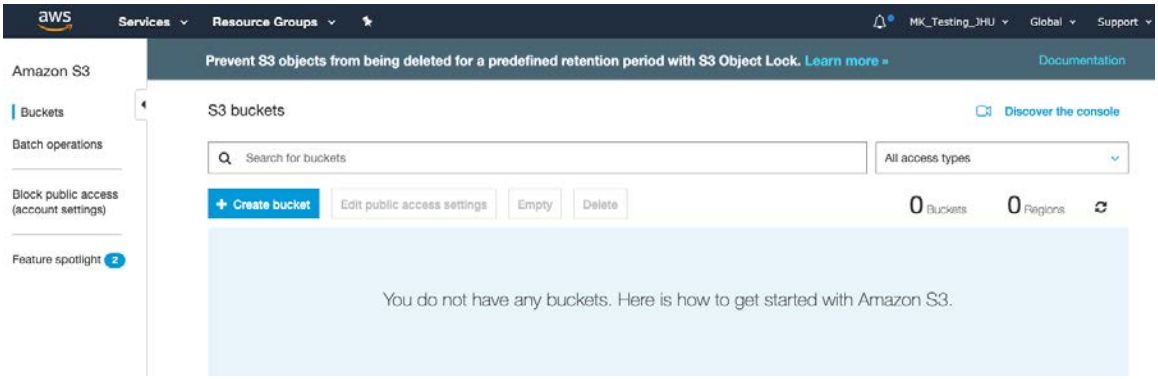
Figure 1: Services Tab[40]



Once the page navigates the user to the Amazon S3 page, the user can then proceed to create a bucket by clicking on the button shown in Figure 2.

[40] Source Minsoo Kim, Figure 1: Services Tab

Figure 2: Amazon S3 Main Page[41]



2.2. Bucket Creation

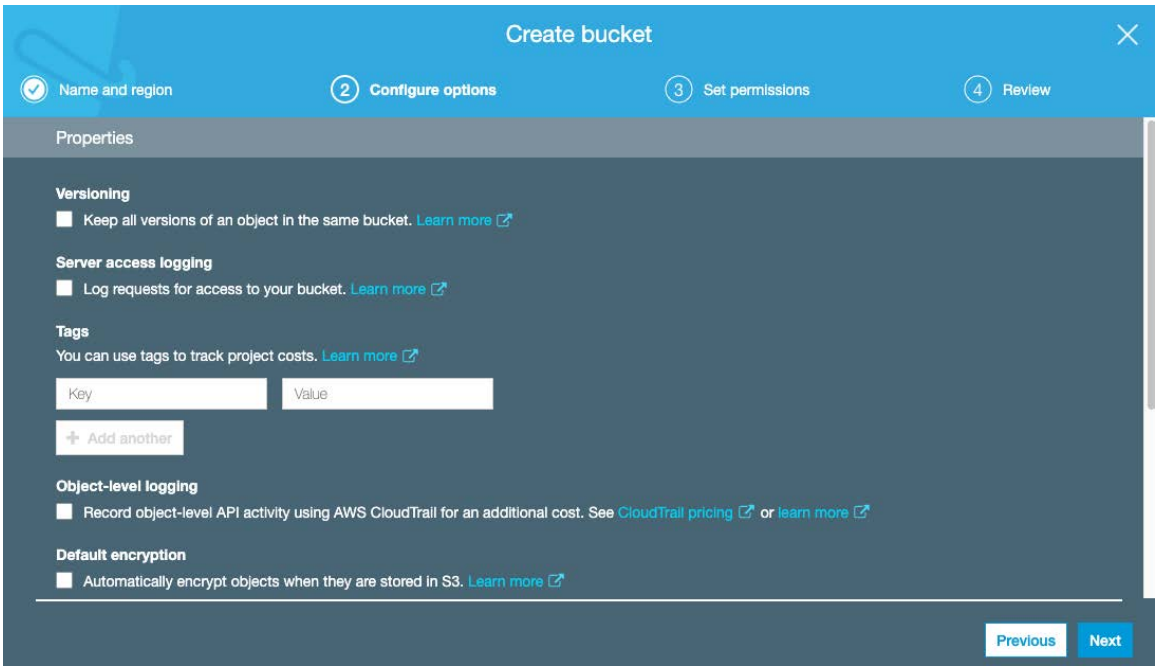
The bucket in Amazon S3 is another terminology for a folder that can be created on the desktop of a computer to store all the files. Buckets act as a folder in Amazon S3. The S3 uses buckets because materials that are stored in the buckets are referred to as "objects." Oddly enough, these objects can be grouped and create a "folder." Therefore, the bucket is the folder that contains "folder," which contains "objects." However, the bucket can contain just the objects if they are not grouped into a folder.

The bucket name, region, and settings from an existing bucket can be selected. Since this is the first bucket being created, the settings from an existing bucket option should be empty. However, the user can utilize the settings from the existing bucket instead of making the same settings for every bucket. Once the information has been filled out, click on the next button to proceed to further configurations. Next step, the bucket's configuration options can be selected. From Figures 3 and 4, the following

[41] Source Minsoo Kim, Figure 2: Amazon S3 Main Page

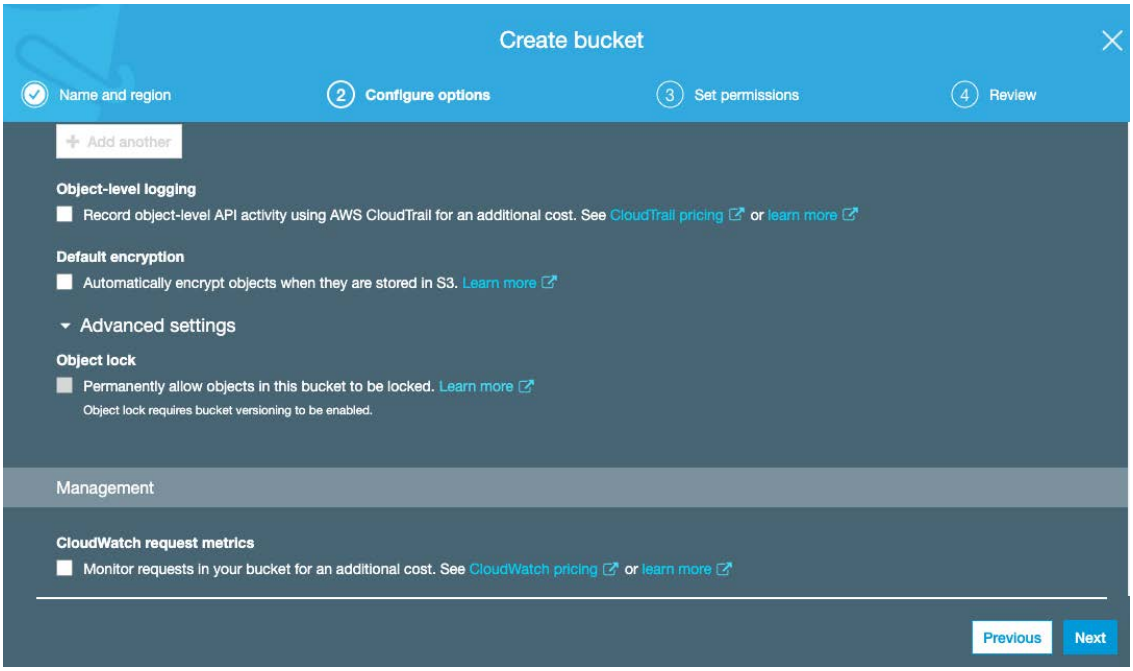
properties are visible: versioning, server access logging, tags, object-level logging, default encryption, object lock, and CloudWatch request metrics.

Figure 3: Create Bucket Part I[42]



[42] Source, Minsoo Kim, Figure 3: Create Bucket Part I

Figure 4: Create Bucket Part II[43]



The versioning feature allows the user to keep all versions of the object uploaded in the bucket. Meaning, if the bucket went through any changes, the versions of the bucket before changes would be kept. All of the previous versions are kept and can be accessed by users. The server access logging allows the S3 to keep log requests for access to the bucket by all users. By turning on this feature, institutions can keep track of who last acquired access to the bucket. Tags are ways to label the buckets so that users can keep track of which bucket is. Object-level logging is a similar feature as the server access logging. The object-level logging keeps a tab on who tried to access the objects within the bucket. The default encryption feature refers to how the uploaded object would be encrypted while being stored in the S3 bucket. Under the advanced settings, the object lock feature "permanently allow objects in the bucket to be locked."[11] This feature prevents the "objects from being deleted to help ensure data integrity and regulatory

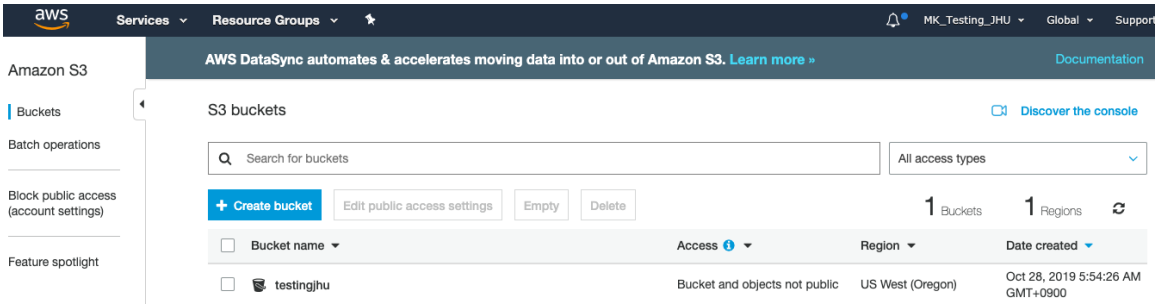
[43] Source, Minsoo Kim, Figure 4: Create Bucket Part II

compliance."[11] The object lock feature comes in handy to make sure that no user can accidentally tamper with the uploaded objects.

Lastly, the CloudWatch requests that metrics refer to another service, AWS CloudWatch. AWS CloudWatch monitors activities that happen within the bucket. If an unusual attempt to access the object in the bucket occurs, the administrator of the account will be notified of such an attempt. Not all features are free to use, but depending on the needs of the institution, these features can be utilized.

Once the configurations have been established, bucket permission settings need to be established before a bucket is created. The general advice is to block all public access, which denies external parties from accessing the bucket and its contents. For safety, the best practice is to avoid anyone from accessing the bucket. Instead, give permissions to the users who need to access the bucket. The setup for creating a bucket has been completed, and a bucket can be located in the Amazon S3 page, as seen in Figure 5.

Figure 5: Successful Bucket Creation[44]

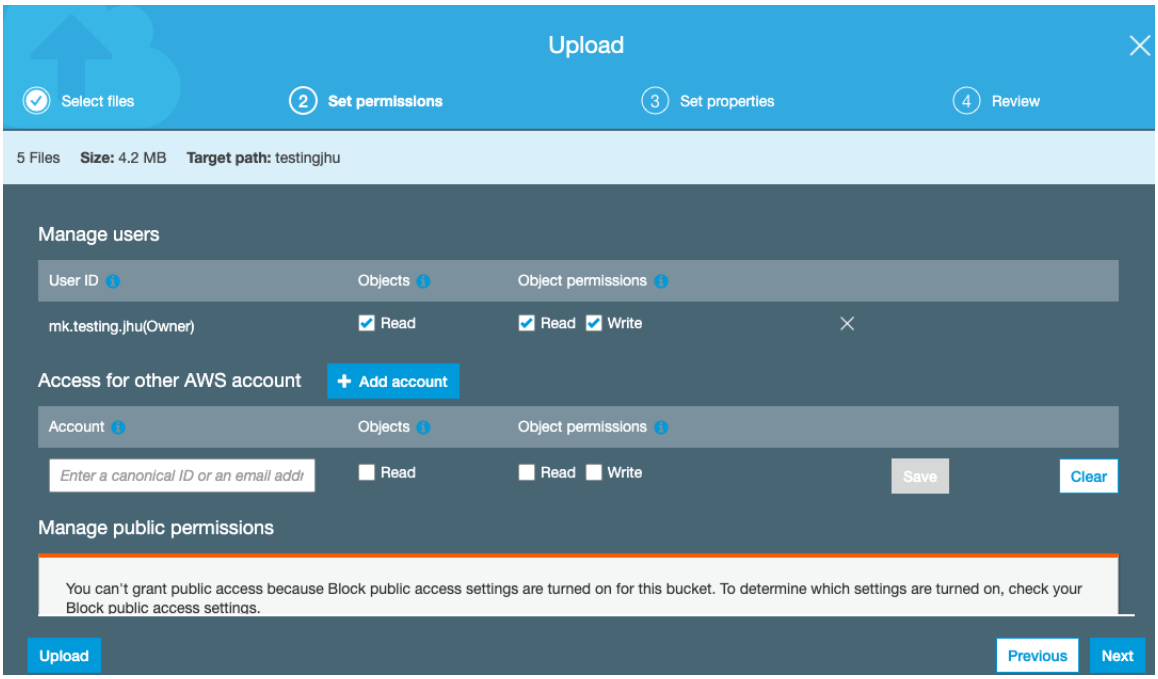


[44] Source, Minsoo Kim, Figure 5: Successful Bucket Creation

2.3. Uploading Objects into the Bucket

Now that the bucket has been created, documents can be uploaded as objects into the bucket. To upload, click on the upload button, and the user can add files to upload to the bucket. Multiple files can be selected and uploaded at the same time. Similar to the bucket creation process, objects require permission settings before the upload. Instead, shown in Figure 6, object permissions request the user to set access for other AWS account. Meaning that adding other users' AWS accounts allow those users to read or write the uploaded object.

Figure 6: Object Upload Settings[45]



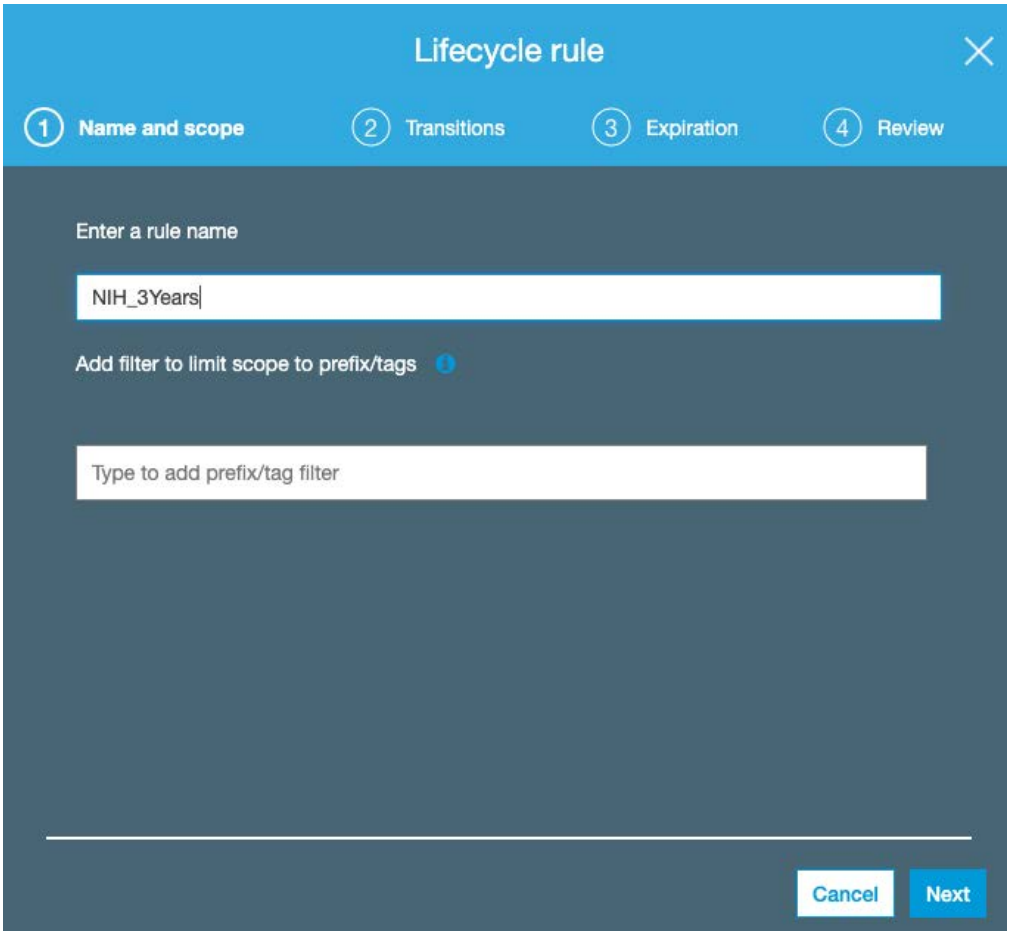
[45] Source, Minsoo Kim, Figure 6: Object Upload Settings

Lastly, the storage class needs to be established before uploading the objects into the bucket. Storage class refers to the tiers designed by the Amazon S3. Based on the use case of the uploaded objects and the frequency of the access, the appropriate storage class can be determined. It is essential to select the most appropriate storage class since the storage class has different price rates. Once the storage class has been selected, the objects are ready to be uploaded in the bucket.

2.4. Lifecycle Policy: Establishing Expiration

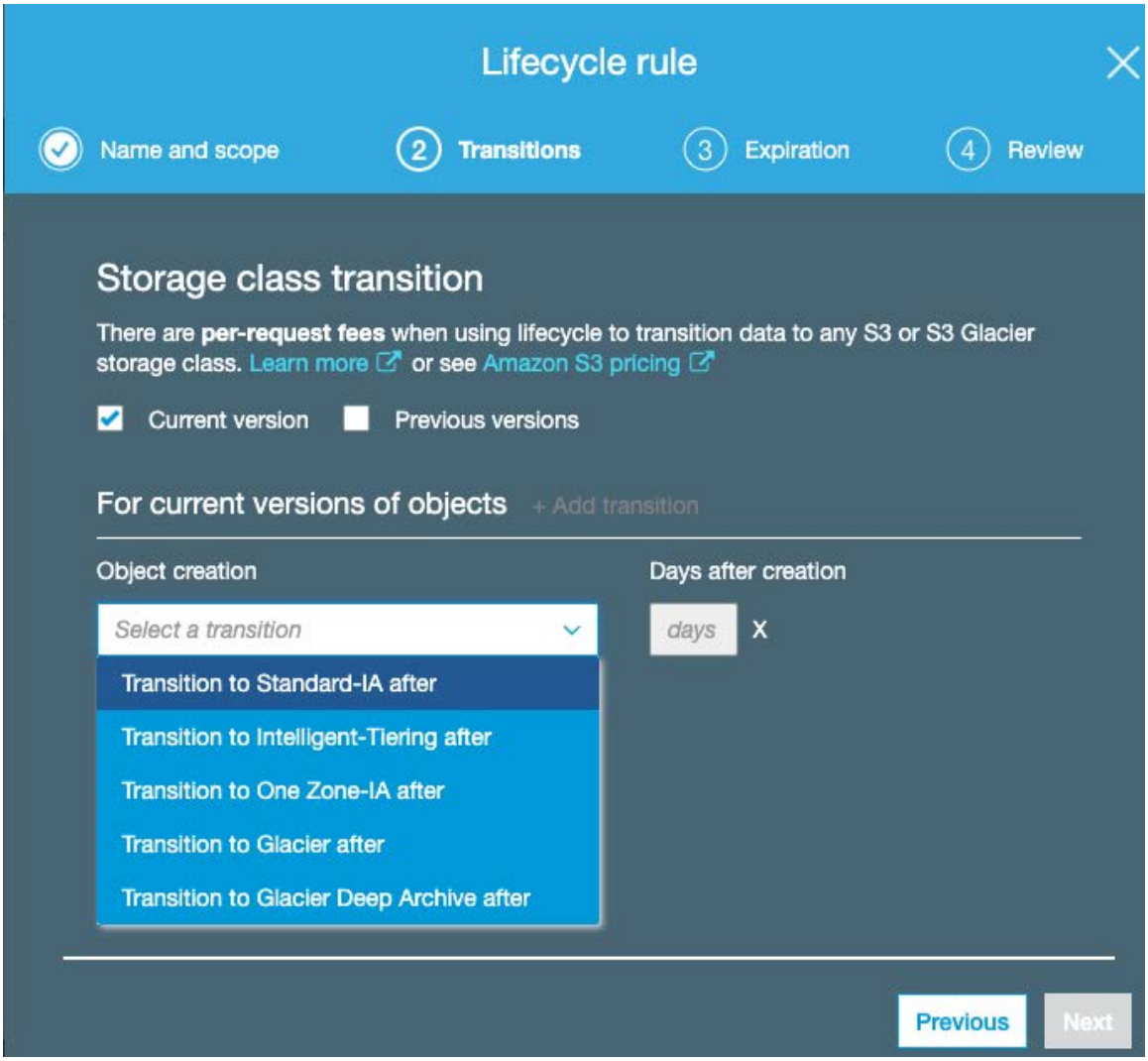
A cloud-based repository has been successfully created. However, there is an essential feature that needs to be addressed. Under the management tab in the bucket, the user can add a lifecycle rule to the bucket. A lifecycle refers to the period in which the bucket will exist in the Amazon S3 repository. Lifecycle policy can either terminate the bucket upon expiration or change the storage class of the objects in the bucket. By changing the storage class, the user can save more money by using a lower storage class tier. Setting a lifecycle policy rule is similar to that of the previous setup of uploading objects. Name the lifecycle rule, select storage class transition or configure expiration date, and complete the lifecycle rule setup. Figures 7, 8, and 9 illustrate the process of setting up a lifecycle rule on the Amazon S3 bucket.

Figure 7: Lifecycle Rule Part I[46]



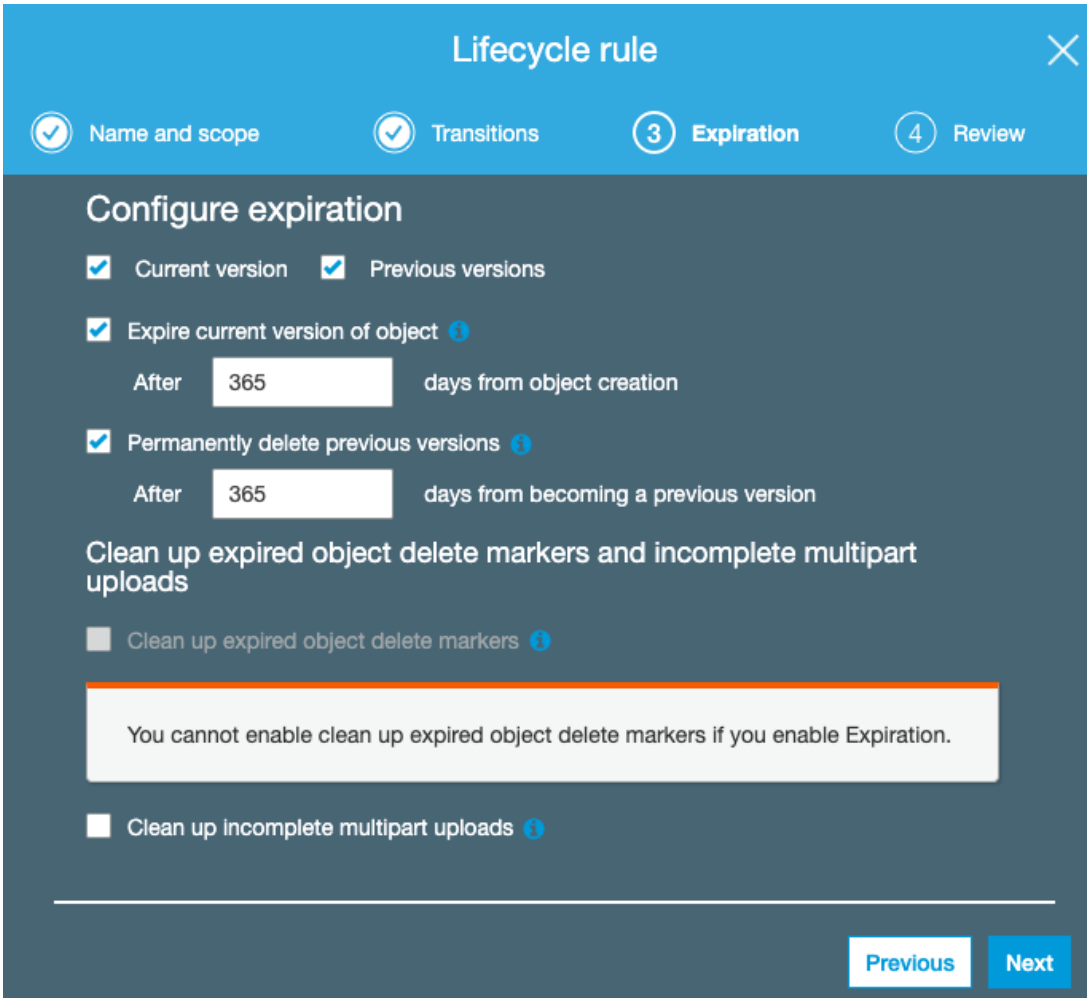
[46] Source, Minsoo Kim, Figure 7: Lifecycle Rule Part I

Figure 8: Lifecycle Rule Part II[47]



[47] Source, Minsoo Kim, Figure 8: Lifecycle Rule Part II

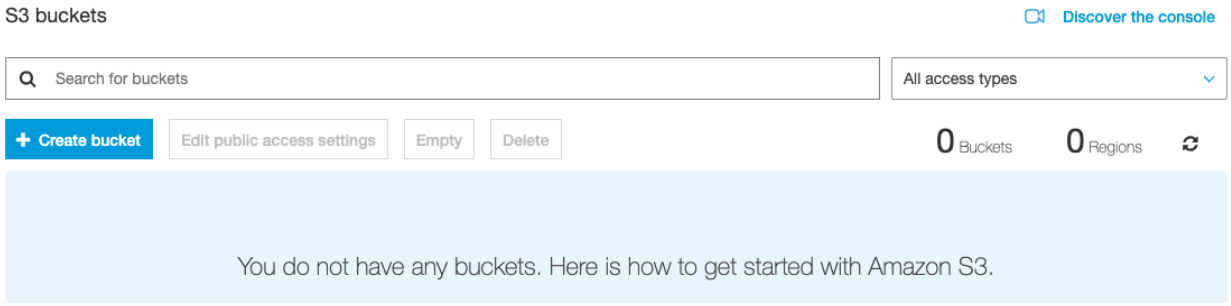
Figure 9: Lifecycle Rule Part III[48]



The following, Figure 10, depicts what the S3 main page would look like if the bucket were terminated by the lifecycle policy. AWS S3 would not contain any bucket since there was only one bucket in the repository and that bucket was terminated.

[48] Source, Minsoo Kim, Figure 9: Lifecycle Rule Part III

Figure 10: Lifecycle Policy in Effect[49]



With the added lifecycle policy, the institution can easily create a cloud-based repository to store documents that arose from research activity and terminate them upon expiration. Furthermore, if the user wanted to share the objects in the bucket, the user could allow access for users with AWS accounts.

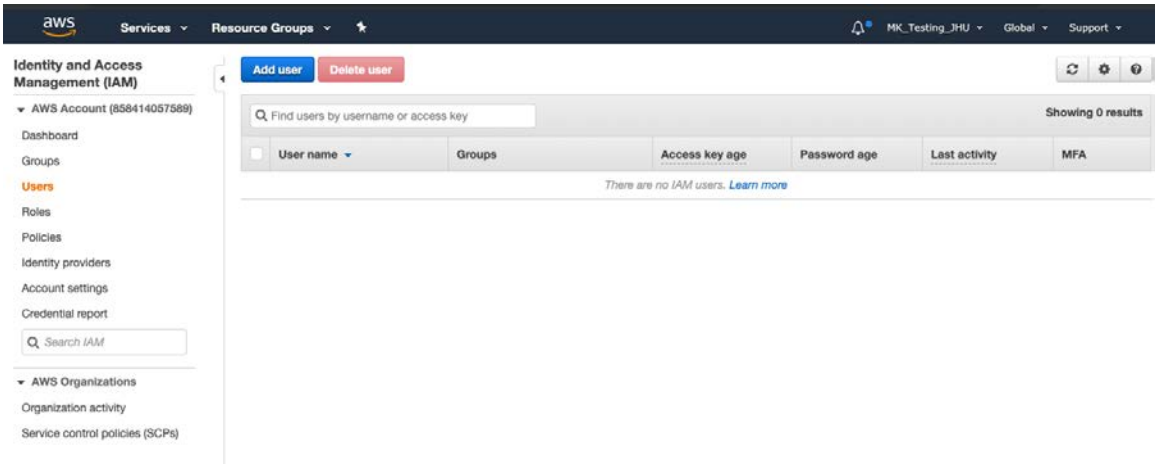
[49] Source, Minsoo Kim, Figure 10: Lifecycle Policy in Effect

Chapter 3. Managing User Access

While the cloud-based repository has been created, the institution faculties cannot share a single account to access the repository. The research administrators should manage access to the cloud-based repository. To manage the user access of the cloud-based repository, the institution can utilize the service known as IAM Users. The Identity and Access Management (IAM) is a tool that allows the root administrator of the AWS account to grant or take away the permission of accessing services provided by AWS. By default, any new IAM user created has no access to any AWS services. Therefore, permissions, via access policy, needs to be provided by the administrator.

Managing user access is as simple as creating the cloud-based repository using Amazon S3. In the AWS Management Console Page, click on the "IAM" under the Security, Identity, & Compliance section. After being directed to the IAM page, click on the "Users" section on the left side of the panel. From there, click on the button, "Add User." The following screen should resemble that of Figure 10.

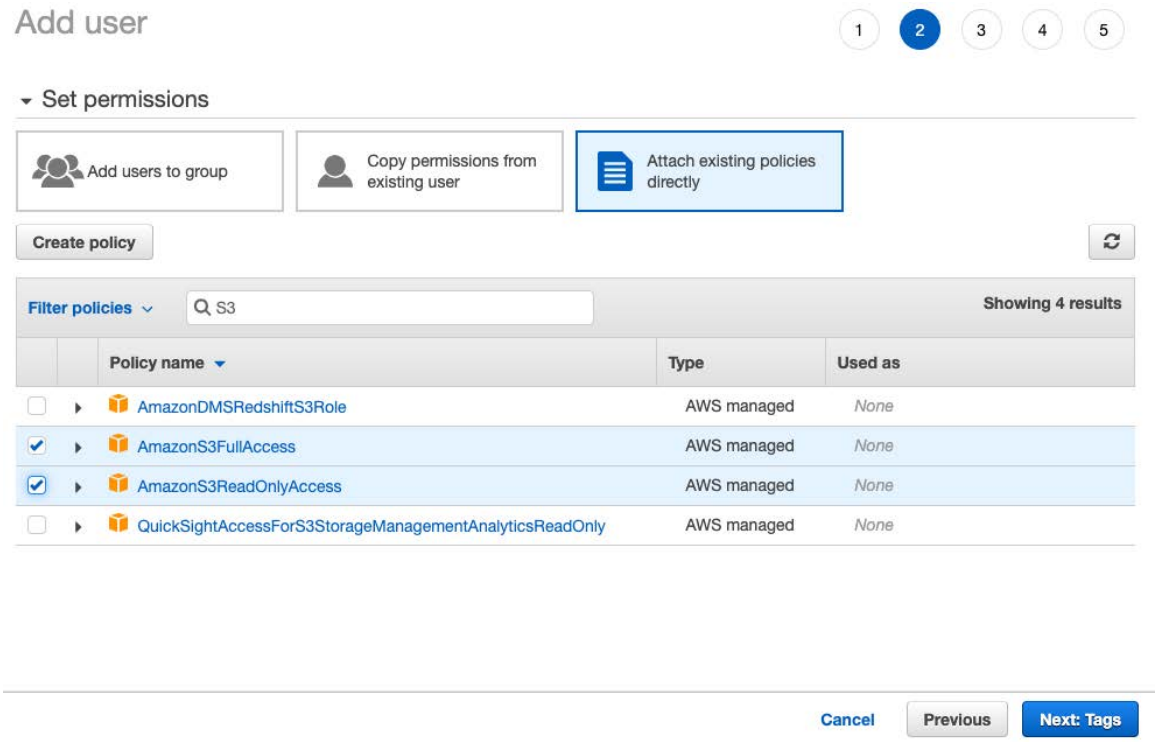
Figure 11: Add User Page[50]



[50] Source, Minsoo Kim, Figure 11: Add User Page

Multiple users can be added and should be granted an AWS Management Console Access type. Console password can be autogenerated and have the user reset the password upon the first login. Next, users can be assigned to existing policies directly. As shown in Figure 11, type in "S3" and filter the policies to attach to the added users.

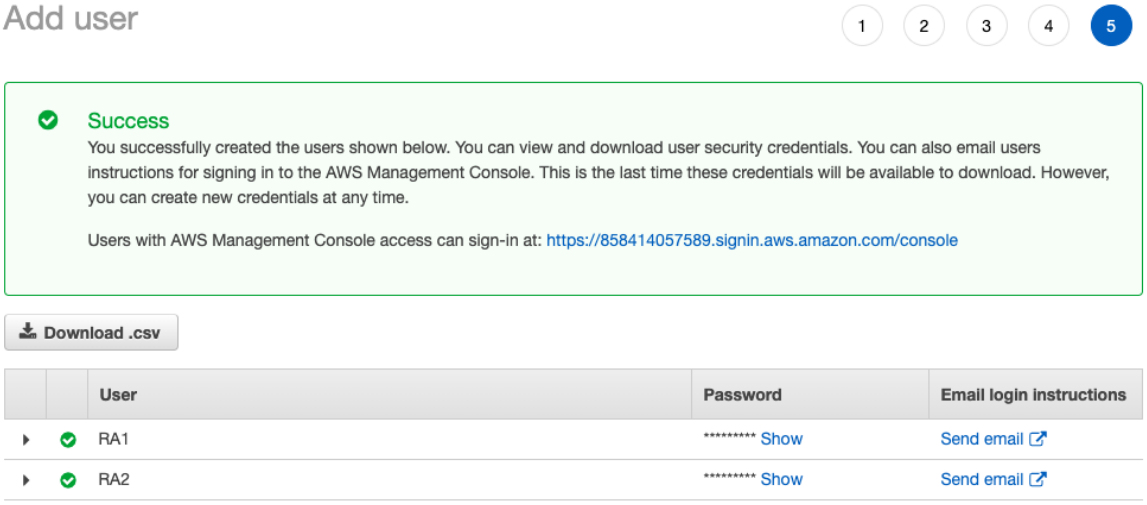
Figure 12: Access Policy[51]



Depending on the added users, full access or read-only access can be attached to the added users. Once the policy has been attached, review and the users are now successfully added to the IAM. On the success page, Figure 12, login instructions can be sent to the added users via email, and the added users can access the Amazon S3 per email instructions that they will receive.

[51] Source, Minsoo Kim, Figure 12: Access Policy

Figure 13: Success Page[52]



[52] Source, Minsoo Kim, Figure 13: Success Page

Chapter 4. Distribution Methods

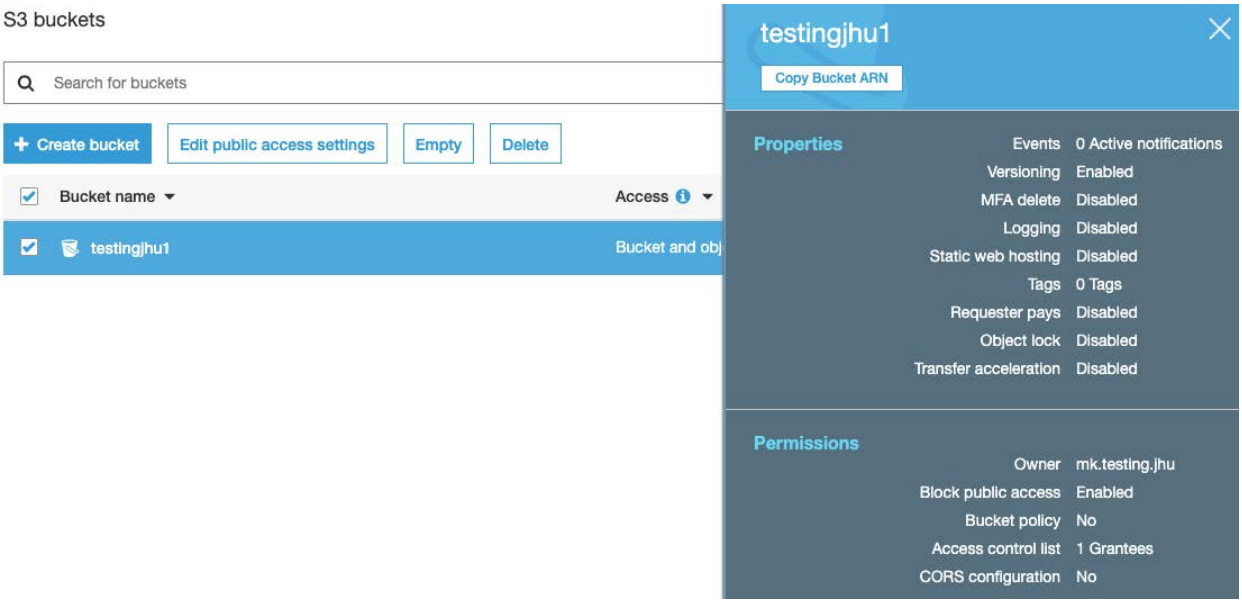
4.1. Bucket Configurations

Once the documents from the research activity have been successfully stored in the cloud-based repository, AWS S3, the documents may need to be accessed by non-faculty members. The retrieval of the documents may be requested due to various reasons: internal audits, research collaboration, and research integrity investigation. The cloud-based repository is created in a private environment – meaning that it is not visible to anyone unless permitted by the administrator. However, there are ways for administrators or users with access to share the uploaded objects with those who do not have access.

It is crucial to note that, without taking the following step, repository contents cannot be shared publicly with anyone without access to the repository. The bucket that contains objects must be made publicly accessible first on the bucket-level. If the bucket's public access setting is not granted, then the bucket and its contents cannot be publicly available. Therefore, if the user has shared the bucket, but the outside recipient cannot see the contents, then the user must confirm that the bucket has been granted public access. It is important to note that by default, the bucket blocks public access upon creation. Therefore, the bucket's public access setting needs to be edited.

Shown on the following Figure 14, the user can see all the created buckets and their properties once the user selects the desired bucket. Under the permissions section, Figure 14 shows that the 'Block public access' has been enabled. Therefore, the bucket blocks all public access and prevents anyone from seeing its contents. If the bucket is made publicly available, the 'Block public access' property would state 'disabled.'

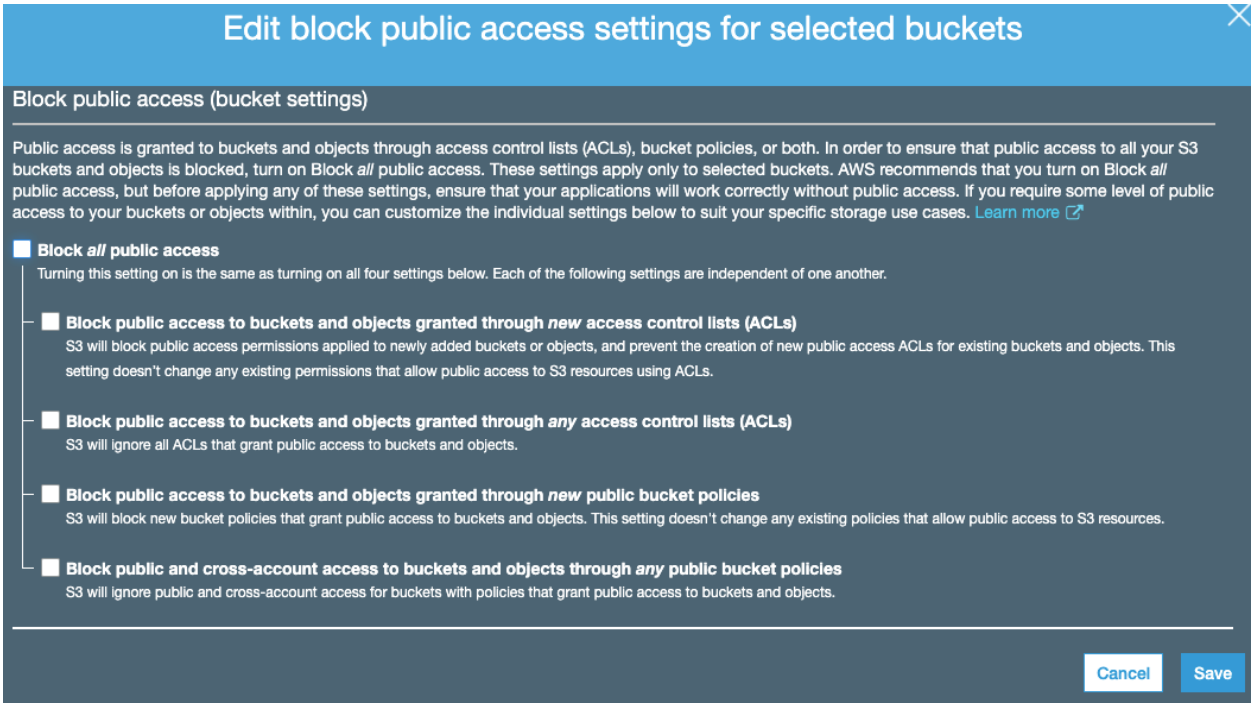
Figure 14: Bucket Properties[53]



To make the bucket available, click on the 'Edit public access settings' button next to the '+ Create bucket' tab. Once the user clicks on the settings tab, the user can uncheck the 'Block *all* public access' box, as seen in Figure 15. The user must save the adjusted settings to ensure the bucket can be shared publicly.

[53] Source, Minsoo Kim, Figure 14: Bucket Properties

Figure 15: Bucket Public Access Settings[54]

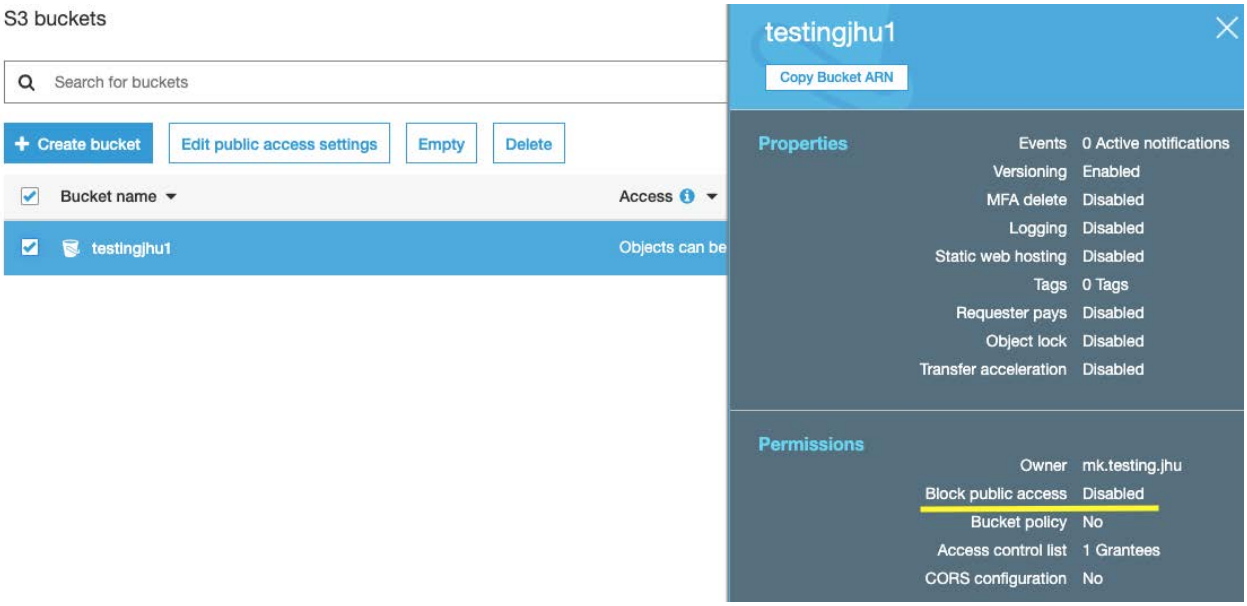


4.2. Confirmation of Changed Bucket Configurations

Figure 16 shows that the following changes made by the user are effective. Additionally, the user can note that the 'block public access' setting is disabled, highlighted by the yellow line in Figure 16.

[54] Source, Minsoo Kim, Figure 15: Bucket Public Access Settings

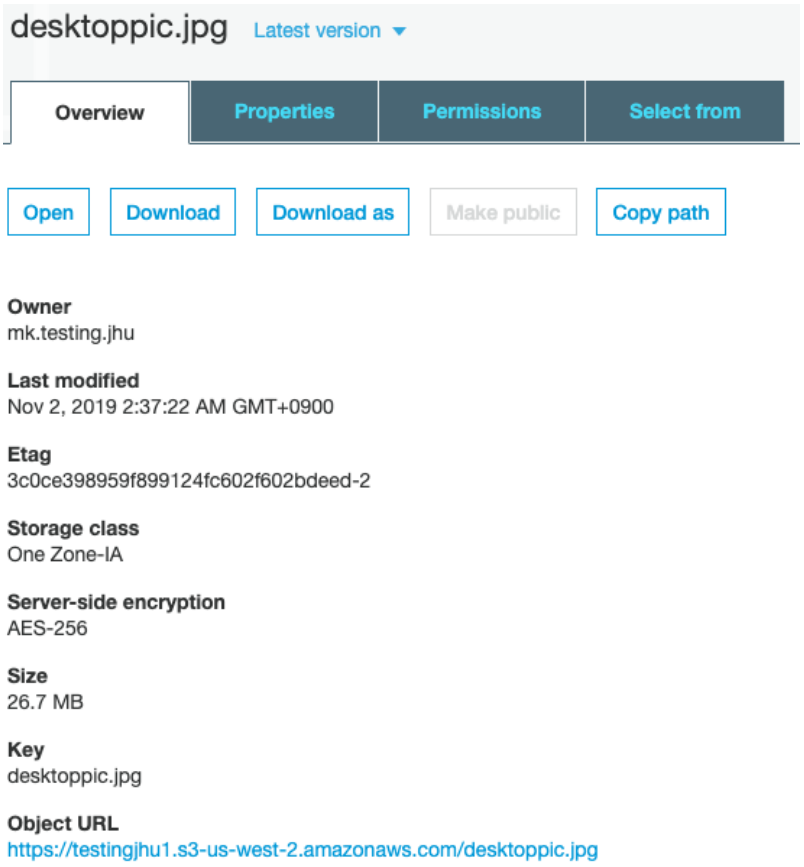
Figure 16: Confirmation of Bucket Public Access[55]



Now that the bucket grants public access, the user can select the objects inside the bucket to share with others. If the user clicks on the object, the user can see the property details of the object illustrated in Figure 17.

[55] Source, Minsoo Kim, Figure 16: Confirmation of Bucket Public Access

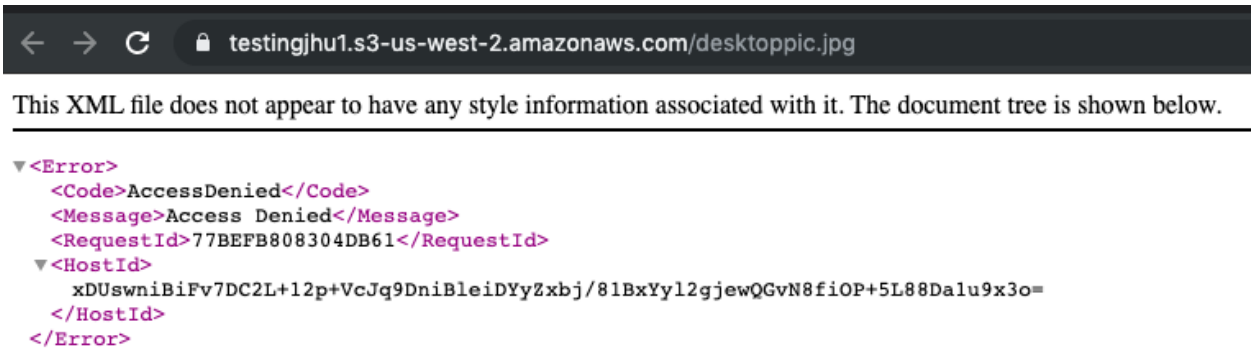
Figure 17: Object Properties Overview[56]



To share the object, the user needs to copy the object URL and send it to the designated recipient. The user can do the same for other objects that need to be shared with external parties. If the bucket were successfully made publicly available, the recipient would be able to view the contents of the bucket. However, if the bucket were not made publicly available, the recipient would see a screen resembling that of Figure 18.

[56] Source, Minsoo Kim, Figure 17: Object Properties Overview

Figure 18: Object Access Denied[57]



Once the contents have been shared and accessed by recipients, the user needs to ensure that the bucket blocks public access again to protect contents in the bucket. If the bucket is not made private, the bucket becomes vulnerable to unwanted exposure to the public. Therefore, it is crucial for users that have access to the repository to ensure that buckets are made private once the sharing has been completed.

[57] Source, Minsoo Kim, Figure 18: Object Access Denied

Chapter 5. Cost Management

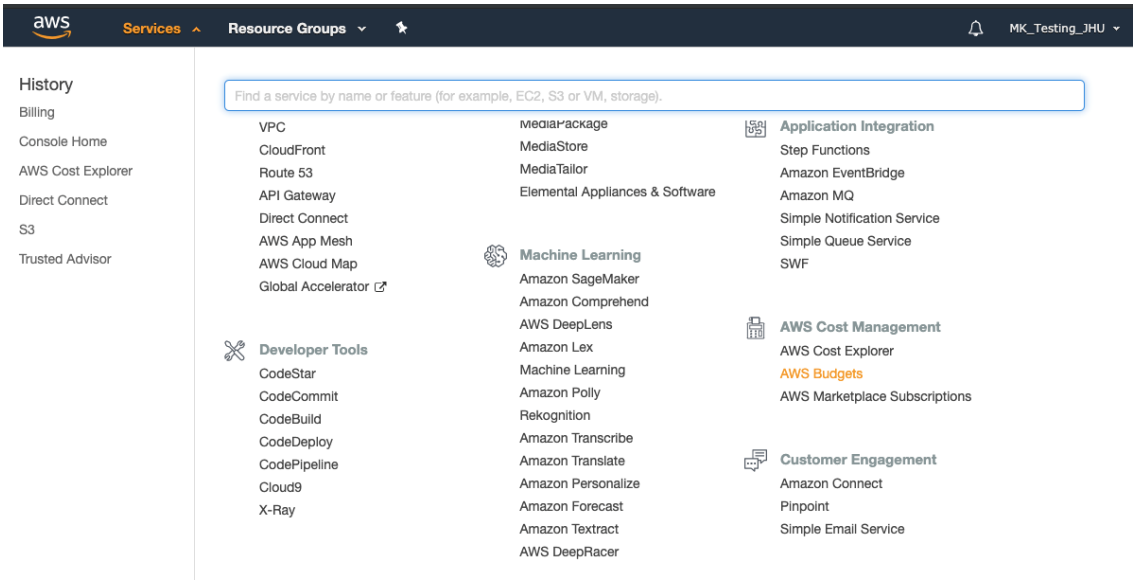
5.1. AWS Budgets

Using a cloud-based repository, institutions can track the financial statements for the usage of products and services in real-time. Furthermore, institutions can implement a threshold to the budget spent on the cloud-based repository. For the cloud-based repository created in this tutorial guide, there are two ways institutions can manage costs: AWS Budgets and AWS Cost Explorer.

AWS Budgets is a service offered by AWS, and it allows users to establish a budget in three different ways based on cost, usage, and reservation. For cost management in a cloud-based repository, reservation method can be neglected. If the institution wants to establish a total budget spent on the cloud-based repository, the institution can utilize the cost budget method to stay under that budget. If the institution wants to ensure that the users do not over-utilize the products and services in AWS and create a huge bill, the institution can put a threshold in the usage budget method to prevent overspending by users.

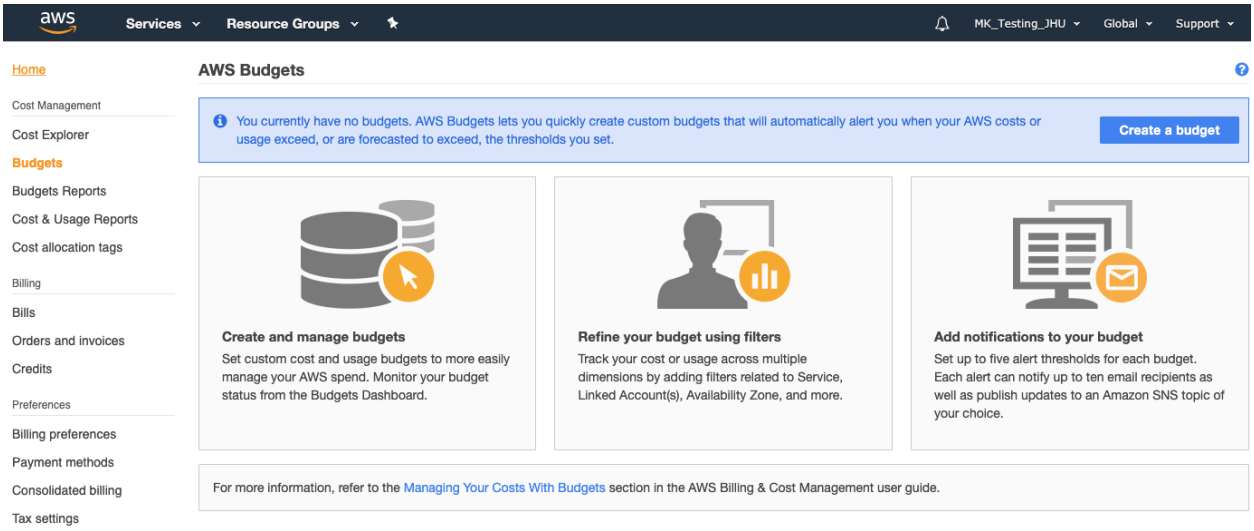
To establish the budget, click on the ‘Services’ tab on the top of the main page after logging in to AWS. Under the ‘AWS Cost Management’ category, select the highlighted ‘AWS Budgets’ illustrated in Figure 19.

Figure 19: AWS Services Tab[58]



The user will be directed to the AWS Budgets page shown in Figure 20. To create a budget, click on the 'create a budget' tab highlighted in blue.

Figure 20: AWS Budgets Main Page[59]

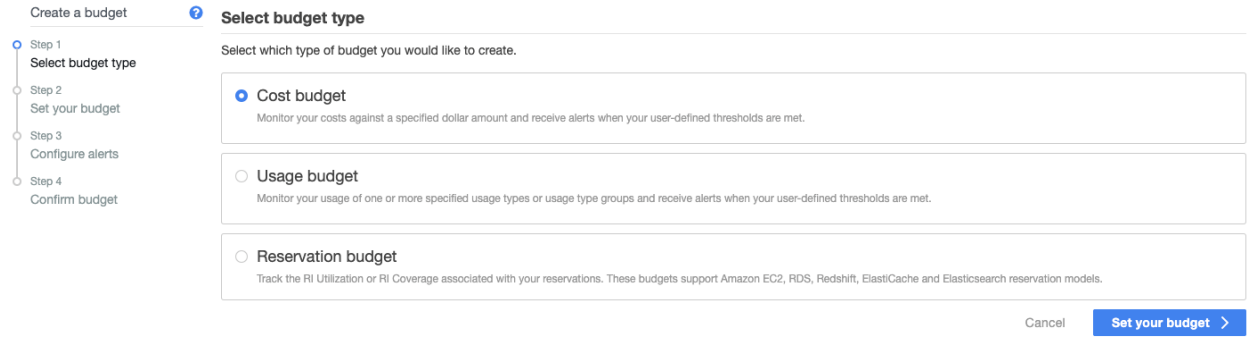


The user can select to establish a budget from the three methods previously mentioned: cost, usage, and reservation. The cost budget method allows the user to put a

[58] Source, Minsoo Kim, Figure 19: AWS Services Tab
[59] Source, Minsoo Kim, Figure 20: AWS Budgets Main Page

total amount spent on the products and services in AWS. The usage budget method establishes a monetary threshold for a specific product and service in AWS. The three budget types are depicted in Figure 21.

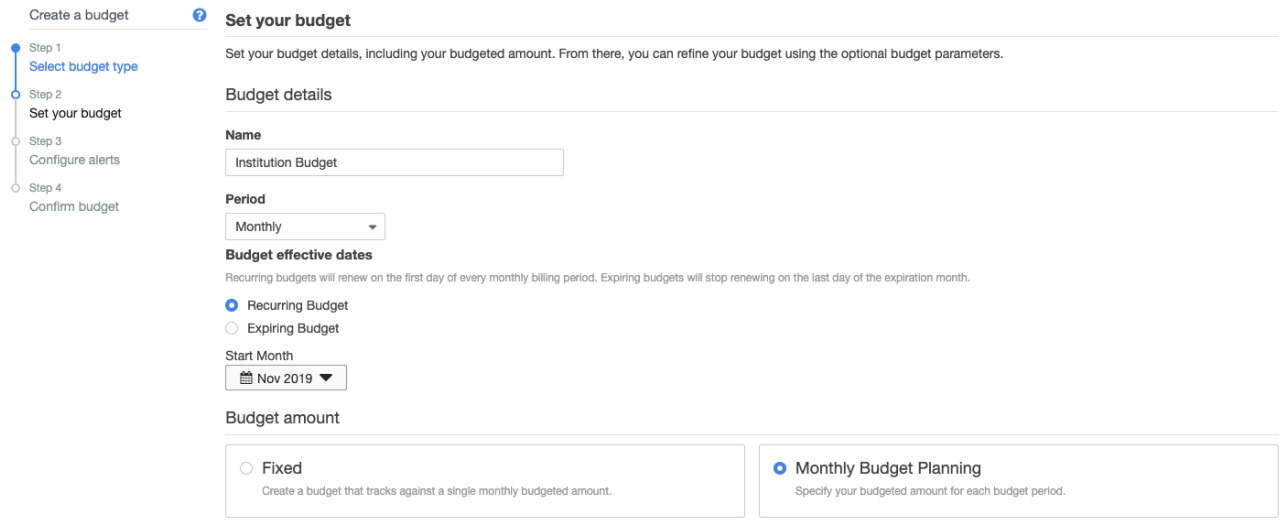
Figure 21: AWS Budget Types[60]



For the demo purpose, the tutorial guide selects a cost budget type method. Shown in Figure 22, the user needs to fill out the name of the budget, period, budget effective dates, and the budget amount. The user can name the budget to remember the type of budget that was established. The period section establishes how frequently – monthly, quarterly, and annually –the budget should occur. The budget effective dates allow users to implement a budget ahead of time. Furthermore, the user can recur a budget or expire a budget after one execution.

[60] Source, Minsoo Kim, Figure 21: AWS Budget Types

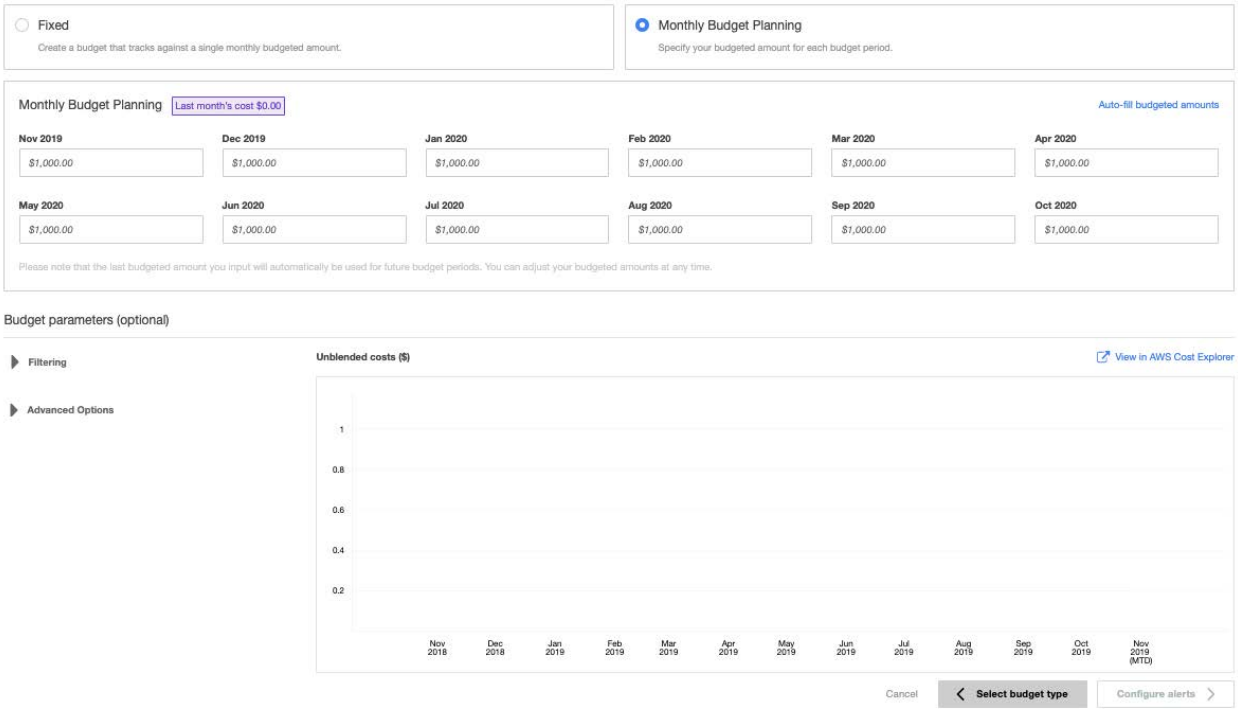
Figure 22: Budget Properties I[61]



Lastly, the budget amount can be established in two ways: fixed and monthly budget planning. Unlike a fixed budget, monthly budget planning lets the user decide if the budget can either increase or decrease over time. If the user selects the monthly budget planning option, the webpage allows the user to manually input the amount of monthly budget in the coming months. The layout of the monthly budget planning can be seen in Figure 23.

[61] Source, Minsoo Kim, Figure 22: Budget Properties I

Figure 23: Budget Properties II[62]

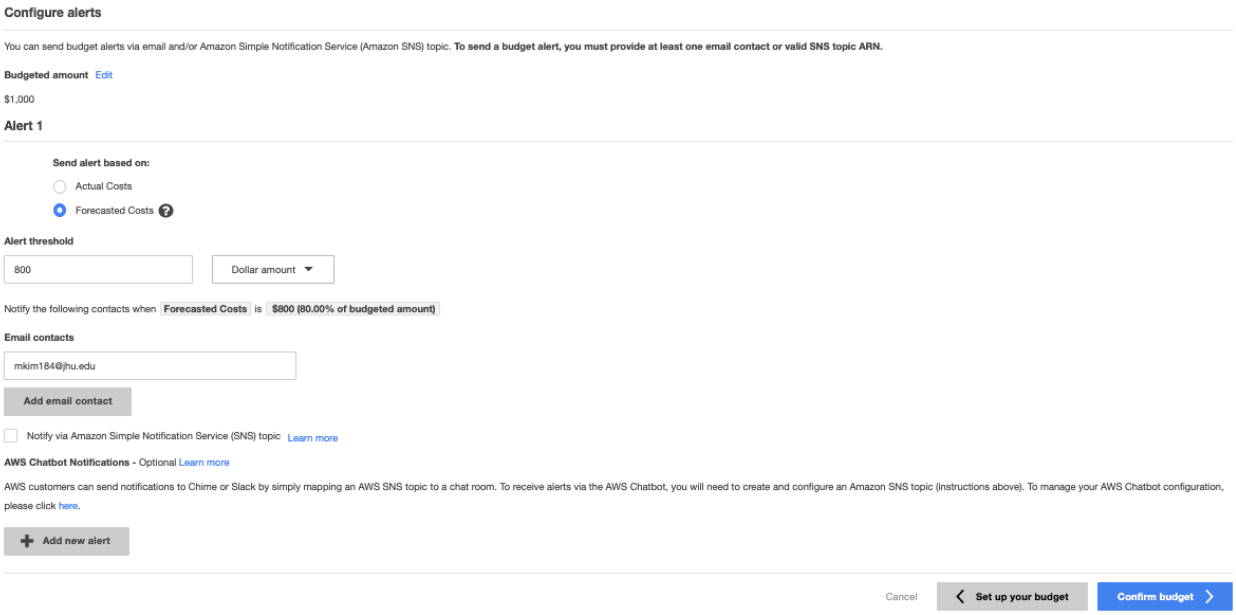


5.2. AWS Budget Notifications

Once the budget type has been created, the user must configure alerts in case the financial expenditure is getting close to the established budget. As seen in Figure 24, the user can configure alerts based on the actual costs or forecasted costs. The user can further establish the alert threshold to receive an alert if the cloud-based repository bill is too close to the threshold. Lastly, the user can add email contacts that can receive alerts and establish more than one alert for the budget.

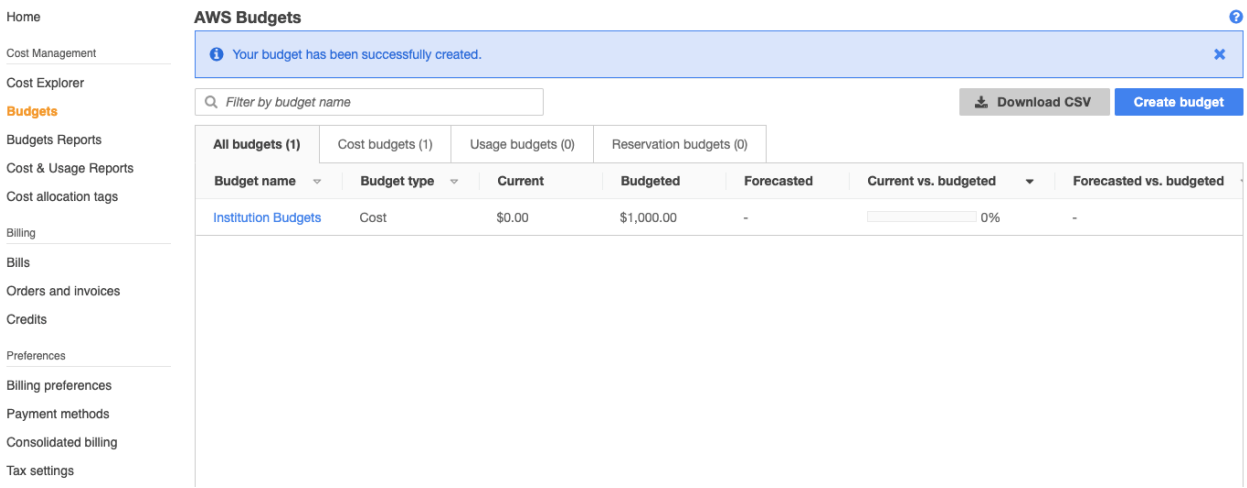
[62] Source, Minsoo Kim, Figure 23: Budget Properties II

Figure 24: AWS Budget Alert Configuration[63]



Once the settings have been filled out, the user can select 'confirm budget' and create a budget. Noted in Figure 25, the user has successfully created a cost budget, named Institution Budgets, that has a threshold of $1,000.

Figure 25: Successful AWS Budget[64]

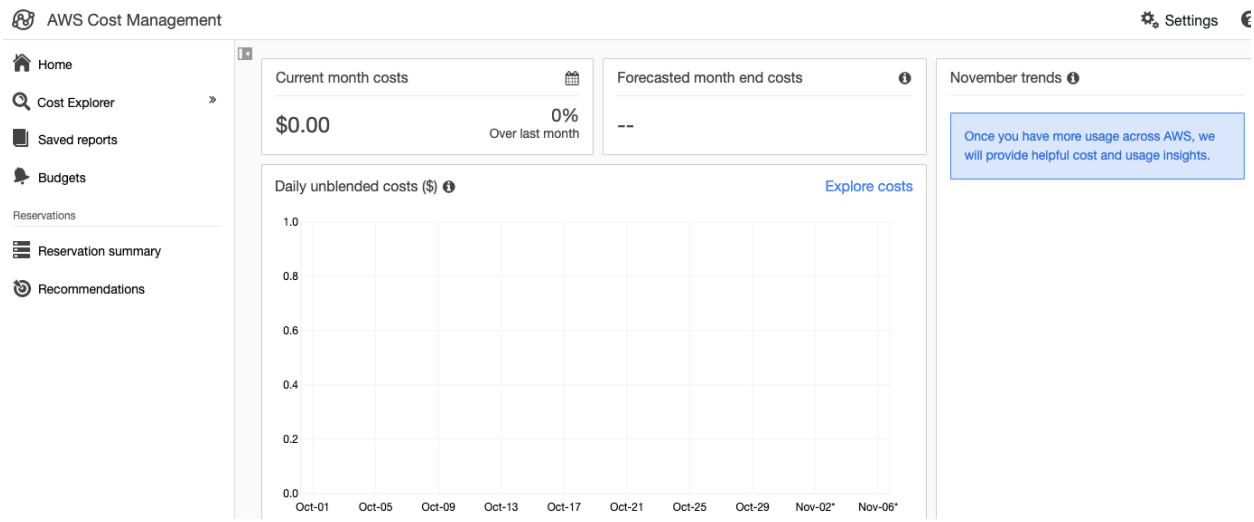


[63] Source, Minsoo Kim, Figure 24: AWS Budget Alert Configuration
[64] Source, Minsoo Kim, Figure 25: Successful AWS Budget

5.3. AWS Cost Explorer

Alternatively, AWS Cost Explorer provides an overview of the total usage of AWS products and services. The significant feature of the AWS Cost Explorer is the predicted cost for the next month. AWS Cost Explorer presents a predicted value of how much the user may utilize in the next month based on the past three months of data. As depicted in Figure 26, the user can see the current month costs, forecasted month end costs, and the chart of daily costs. Figure 26 is for a demonstration-purpose, so it does not show actual numbers spent on the cloud-based repository.

Figure 26: AWS Cost Management[65]



[65] Source, Minsoo Kim, Figure 26: AWS Cost Management

Chapter 6. Recommendations

6.1. Recommendation 1. Use Both Cloud-based Infrastructure and Current System.

While the benefits of using a cloud-based infrastructure may be evident, not everyone keeps up with the change in the system immediately. Additionally, stakeholders can potentially disagree with the necessity to migrate from the current system to a cloud-based infrastructure in order to handle the administrative burden. These are both feasible possibilities that could occur at any institution looking to make a transition into a cloud-based system. For such a case, it is recommended that the institution utilize both the current system and a newly adopted cloud-based infrastructure as a hybrid model.

Despite the benefits of a cloud-based repository, it is possible that the faculties have already adapted to the current system, and it is more difficult for them to try to get used to a newly implemented system. The benefit of adopting a cloud-based infrastructure is the flexibility of using a hybrid model. Ultimately, the institution can use some parts of the current system and handle the rest of the workload with the cloud-based system. The handbook provides a guide to creating a cloud-based repository for the post-award phase; thus, the institution can continue to use the current system for the pre-award and award phases. Institutions can slowly adopt a cloud-based infrastructure to handle both pre-award and award phases in the future.

6.2. Recommendation 2. Customize the Cloud-based Infrastructure to Handle the Complete Award lifecycle.

Institutions use products and services from different vendors that suit their needs in each phase of the award lifecycle. In this handbook, the guide focuses specifically on

creating a cloud-based repository to handle record management in the post-award phase. However, the guide is created on an AWS cloud platform, which provides diverse services to handle the complete award lifecycle. Instead of using multiple software-based solutions to migrate to a cloud-based infrastructure, institutions could focus on utilizing a single platform that can provide needed products and services. AWS is a customizable platform, so it is possible to create a universal platform for all institutions of higher education across the US.

The following is an example of the tasks that can be accomplished using services by AWS to handle pre-award and the award phases:

1. Use Elastic Compute Cloud (EC2) to create a proposal submission system. The system can be web-based and require the faculties to fill-out and submit proposals through the system. Furthermore, the EC2 can be used to create a search engine that can navigate through funding opportunities posted in the federal agencies.
2. Once the proposal has been submitted, Simple Work Flow (SWF) can be administered to track the progress of the proposal submission. SWF is a service that administers the entire workflow of a process. Thus, the SWF can be used to track the progress of the proposal and manage the budget usage of international collaboration.
3. Amazon Simple Notification Service (SNS) can be set up so that the faculty receives notification of the progress or alerts in real-time. Additionally, SNS can be used to inform the faculty of the budget expenditure. If the budget expenditure of an award is reaching 80% of the threshold, the SNS can be triggered to alert the

faculty. Alternatively, the SNS can be triggered to let faculty know that proposal submission requires additional documents.

The example illustrates diverse and sophisticated services that AWS provides for institutions to take advantage of and improve the adopted cloud-based repository to handle the complete award lifecycle.

Glossary

Cloud-based.	On-demand computer system resources, such as applications, services, and storage, for users to access via the Internet.
Cloud-based Repository.	Storage service accessed by users using a cloud-based server.
Cloud Infrastructure.	Hardware and software components that makeup cloud computing.
Content Distribution Network.	Geographically distributed network that delivers or distributes content to various locations.
Cost Optimization.	Ensuring the maximized output for the minimum input required to perform the necessary tasks.
Elasticity.	The ability for the computing system to continually increase and decrease in size to compete with the demanded workload.
Fault Tolerance.	The ability for the computing system to function without delay even if the portion of the system has been impacted.
High Availability.	The ability to create numerous backups of the computing system.
Indirect Costs.	Financial expenditures relevant to the sponsored projects that cannot be directly associated with individual projects. Examples of Indirect Costs include the cost of electricity, administrative services, and usage of facilities. Indirect Costs are additionally known as Overhead or Facilities and Administration (F&A) Costs.
Lifecycle Policy.	Feature that allows modification of the lifespan of stored objects.
Post-Award Phase.	The last phase in the lifecycle of an award, which includes implementation, reporting, and closeout.
Record Management Service.	The storage service that the University of Washington offers to store documents in the post-award phase.

Simple Storage Service (S3).	Object storage designed to store and access any data over the Internet.[66]
S3 Glacier.	Secure, durable, and extremely low-cost Amazon S3 cloud storage classes for data archiving and long-term backup.[67]

[66] Lee Perlitz and Steven G. Elliott. "The Products." Amazon. Pearson Education Australia, 2000. https://aws.amazon.com/products/storage/?nc2=h_ql_prod_st.
[67] Ibid

Abbreviations

AWS	Amazon Web Services
CDN	Content Delivery/Distribution Network
F&A Costs	Indirect Costs, also referred to as Facilities and Administrative Costs
Glacier	Referring to S3 Glacier
IHE	Institutions of Higher Education
S3	Simple Storage Service
TCO	Total Cost of Ownership

References

"Amazon S3 Pricing," Last modified 2002. https://aws.amazon.com/s3/pricing/?nc=sn&loc=4.

Perlitz, Lee and Elliott, Steven G. "The Products." Amazon. Pearson Education Australia, 2000. https://aws.amazon.com/products/storage/?nc2=h_ql_prod_st.

"PeopleSoft Enterprise Grants Management." Oracle. Accessed October 25, 2019. http://www.oracle.com/us/products/applications/peoplesoft-enterprise/service-automation/peoplesoft-grants-management-065800.html.

"PeopleSoft Grants." PeopleSoft Grants - University of Houston, Last modified September 26, 2019. https://www.uh.edu/research/sponsored-projects/peoplesoft/.

"Post-Award Management Software: Monitor and Report Faster." Cayuse. Accessed October 22, 2019. https://cayuse.com/post-award/.

"Research Administration System." The Research Administration System | Controller's Office. Accessed October 25, 2019. https://controller.ucsf.edu/how-to-guides/contracts-grants-accounting/research-administration-system.

Rockwell, Sara. "The FDP Faculty Burden Survey." Research management review. U.S. National Library of Medicine, 2009. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2887040/.

Saas, Tyler Kemp, James Deloitte Consulting LLP, It Takes an Eco-System: A review of the Research Administration Landscape, Research Management Review, Volume 22, Number 1 (2017)

"University of San Diego Case Study." Cayuse. Accessed October 25, 2019. https://cayuse.com/case-study/university-san-diego-case-study/.

Appendix 2: Biography

Minsoo Kim is currently an AWS Consultant for a global company, which provides services to enterprise clients that seek to utilize and migrate their current infrastructures to a cloud platform. He is a certified AWS Cloud Practitioner and AWS Solutions Architect Associate. Before working as an AWS consultant, he had opportunities to work at Apple Inc., in an Apple Developer Program team, and at Amazon, in an Amazon Web Services (AWS) team. Additionally, he has had the opportunity to work at Acuity Polymers, a biomedical company, as a biomedical engineer associate. Apart from his career, he has an educational background in receiving a Bachelor of Science degree in Biomedical Engineering at the University of Rochester. Now he is on his way to completing the Master of Science in Research Administration at Johns Hopkins University.